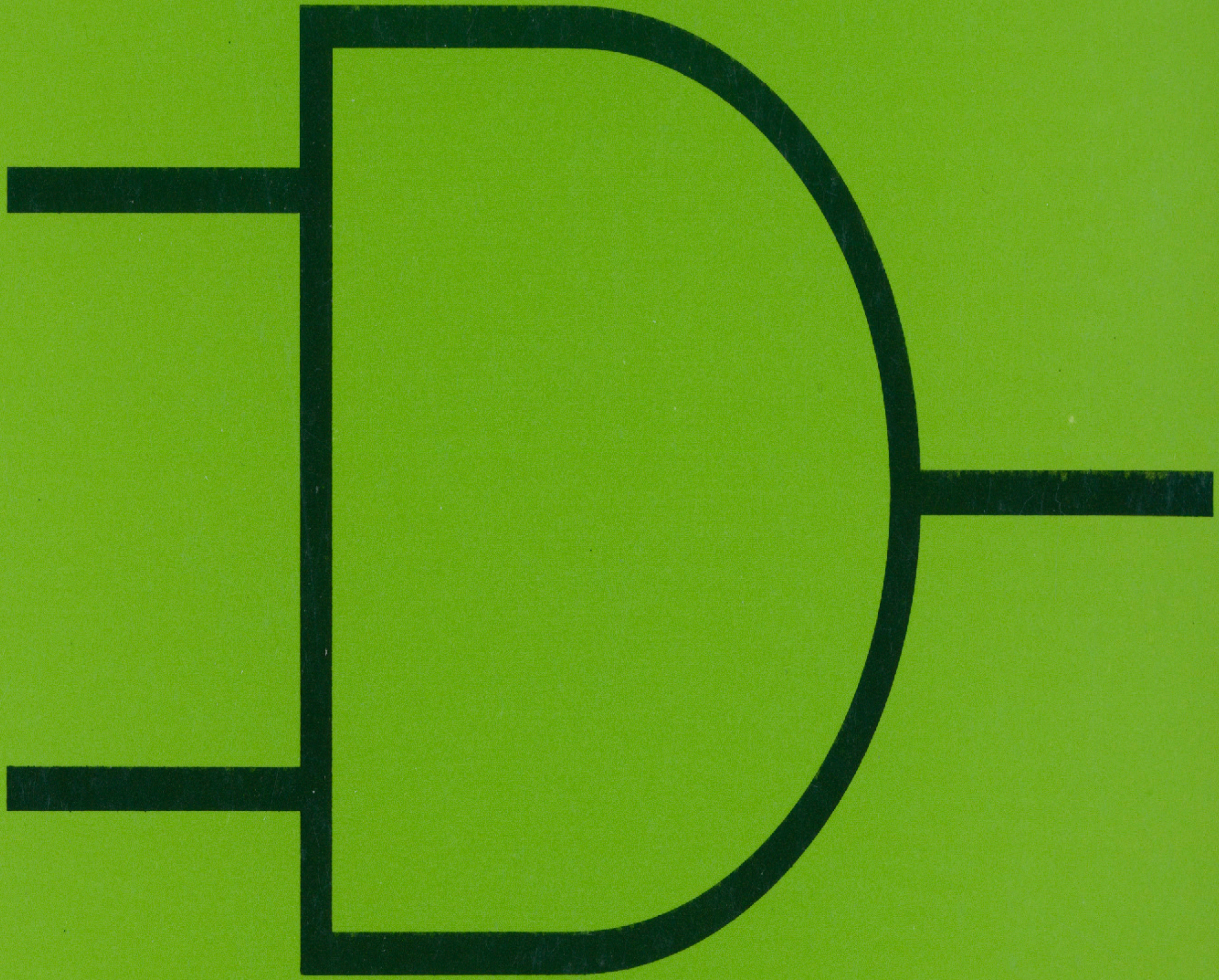
THORN EMI Television Rentals
MACMILLAN
SERIES



Logic gates
Flip-flops
Counters and shift registers

An introduction to digital techniques

THORN EMI Television Rentals Macmillan Series

An introduction to digital techniques books

An introduction to digital techniques
Digital displays and applications
An introduction to microprocessors
VHS recording principles

Tutor boards

Digital tutor board with exercises
Transistor tutor board with exercises

Video tapes

Introduction to microprocessor systems
Introduction to the Teletext system
VHS mechanical systems

For the tutor

Two-day teletext course
Basic VHS course

For everyone

Basic mathematical skills

An introduction to digital techniques

Logic gates, Flip-flops, Counters and Shift registers

MACMILLAN
EDUCATION

THORN EMI
THORN EMI
Television
Rentals Ltd.

© THORN EMI Television Rentals Group Technical Training Department 1977,
1978, 1979, 1984

All rights reserved. No part of this publication
may be reproduced or transmitted, in any form
or by any means, without permission

Part 1 originally published as *Book 1 Logic gates*, 1977
Part 2 originally published as *Book 2 Flip-flops*, 1978
Part 3 originally published as *Book 4 Counters and Shift registers*, 1979

Published in this format in 1984 by
MACMILLAN EDUCATION LTD
Houndmills, Basingstoke,
Hampshire, RG21 2XS,
and London
Companies and representatives
throughout the world

British Library Cataloguing in Publication Data

An introduction to digital techniques.—
(Thorn EMI Macmillan series)
1. Digital electronics
I. Thorn EMI Television Rentals
621.3815′3 TK7868.D5
ISBN 978-0-333-36415-4 ISBN 978-1-349-07077-0 (eBook)
DOI 10.1007/978-1-349-07077-0

Contents

Part 1 Logic gates 1
Electronic gates 3
The OR function 4
The NOT OR function 6
The AND function 8
The NOT AND function 10
The EXCLUSIVE OR function 12
Gates operated by low level input pulses 14
The BUFFER AMPLIFIER function 20
The INVERTING AMPLIFIER function 21
A NOT gate from other gates 22
Summary of gate symbols 24
Revision 26
Solutions to exercises 27
Exercises 28
A synopsis of gate symbols 32

Part 2 Flip-flops 33
The transistor R-S FLIP-FLOP 35
R-S FLIP-FLOPS from logic gates 38
Low input activated FLIP-FLOP 39
High input activated FLIP-FLOP 40
Clocked R-S FLIP-FLOP 45
Clocked 'D' type FLIP-FLOP 46
Edge triggered FLIP-FLOPS 47
The toggle FLIP-FLOP 49
Negative edge triggering 52
The J-K master slave FLIP-FLOP 54
J-K master slave with set-reset facility 57
Appendix to Part 2 61

Part 3 Counters and Shift registers 65
Binary numbers 67
Asynchronous counters 72
Synchronous counters 77
Decade counters 86
Shift registers 88

Glossary of terms 95

Solutions to Exercises 100

Index 102

Part 1

Logic gates

INTRODUCTION

This volume looks at digital techniques and circuitry, relating them to the present state of the art in the television receiver, and also provides an insight into basic functions used in equipment such as remote controls.

Electronic gates

An electronic gate is a device usually made from transistors with two or more inputs and one output.

The gates may be made of ordinary NPN transistors or field effect transistors (FET).

VOLTAGE LEVELS

Gates are operated by input voltages, and may be operated by a 'high' or 'low' voltage level.

The terms 'high' level and 'low' level are explained by the examples in figure 1.1.

The more positive voltage is called the 'high' level and the less positive voltage is called the 'low' level, eg. in Figure 1.1 –5v is more positive than –8v.

GATE SYMBOLS

Two of the most common gates are the *AND* gate which performs the *AND* function and the *OR* gate which performs the *OR* function. The circuit symbols of the *AND* gate and the *OR* gate are shown in figure 1.2.

Fig. 1.1 A comparison of voltage levels.

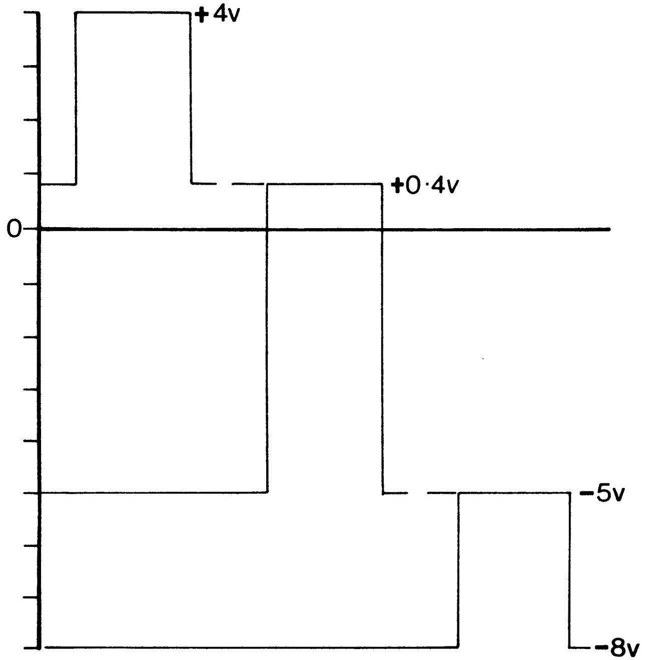
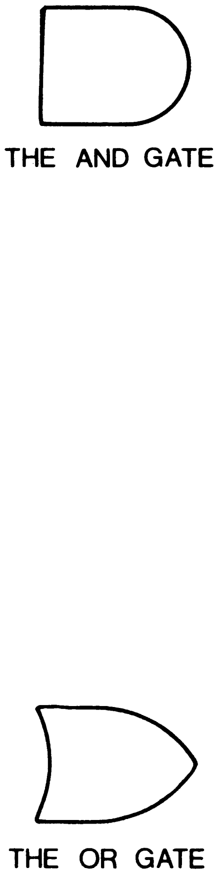


Fig. 1.2 The circuit symbols of the AND gate and the OR gate.



The OR function

In figure 1.4 when switch A *OR* switch B *OR* both switches are closed there will be an output across R. When both the switches are open there will be no output across R.

AN EQUIVALENT CIRCUIT

A simple *OR* gate made with two NPN transistors is shown in figure 1.5. A 'high' level input at A *OR* B *OR* both, would cause one or both transistors to conduct producing a 'high' level output across R.

If both transistors are switched off by a 'low' level input there is no output ('low' level) across R.

The circuit symbol for the *OR* gate just described is shown in figure 1.3.

Fig. 1.3 The OR gate circuit symbol.

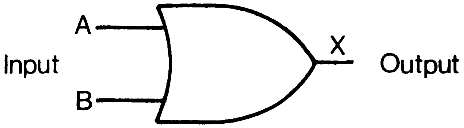


Fig. 1.4 Two switches used as an OR gate.

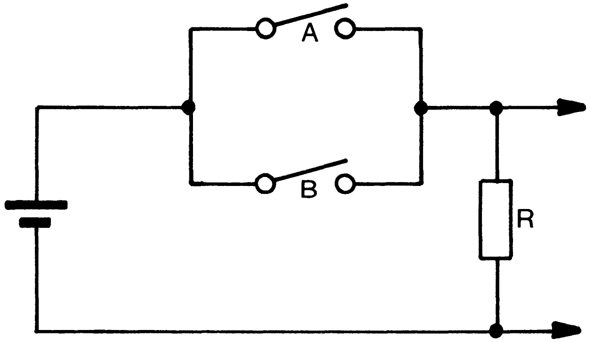
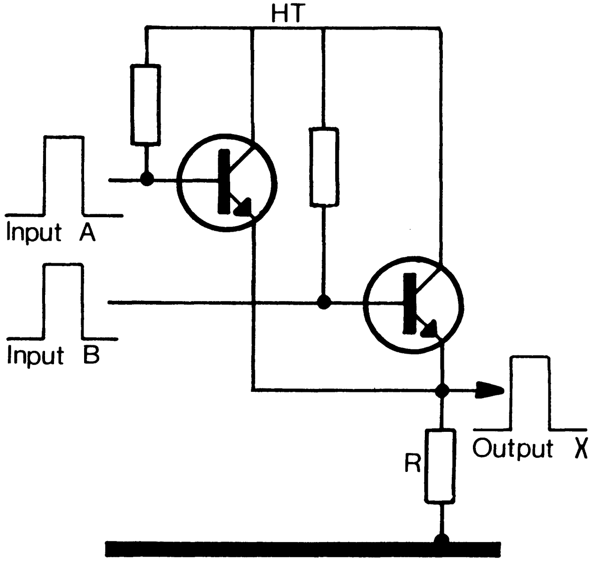


Fig. 1.5 An OR gate from two NPN transistors.



THE VOLTAGE LEVEL TRUTH TABLE

The truth table covers all possible combinations of inputs, and gives the resultant output from each combination. 'H' represents 'high' level and 'L' represents 'low' level.

A PRACTICAL USE OF AN OR GATE

The diagram in figure 1.8 shows a practical use for an *OR* gate. It could be used in a burglar alarm system where the inputs come from doors, windows etc. If a window or door is opened A or B will go 'high' giving a 'high' output at X which would be used to trigger the alarm.

EXERCISE (FIGURE 1.9).

The illustration gives the input levels for A and B of an *OR* gate. Draw in the output levels at X. The solution is on page 27.

Fig. 1.6 The voltage level truth table.

Inputs		Output
A	B	X
H	H	H
H	L	H
L	H	H
L	L	L

Fig. 1.7 Summary of the truth table.

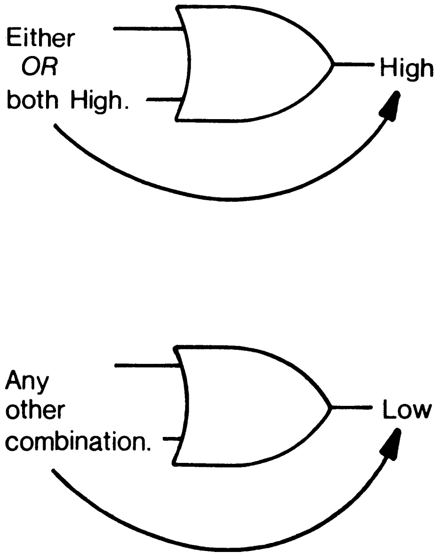


Fig. 1.8 The OR gate used in a burglar alarm system.

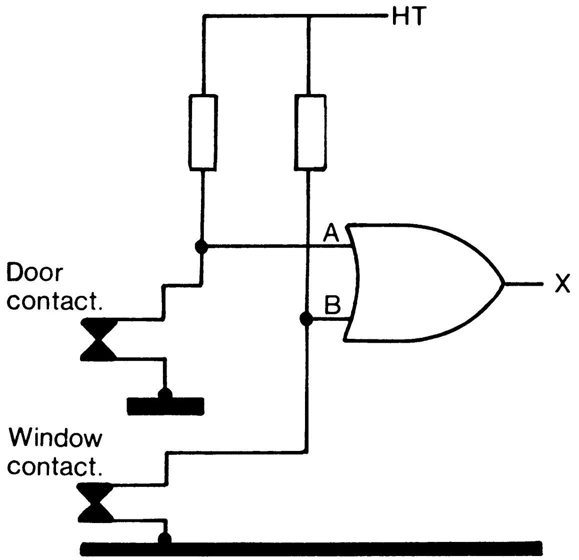
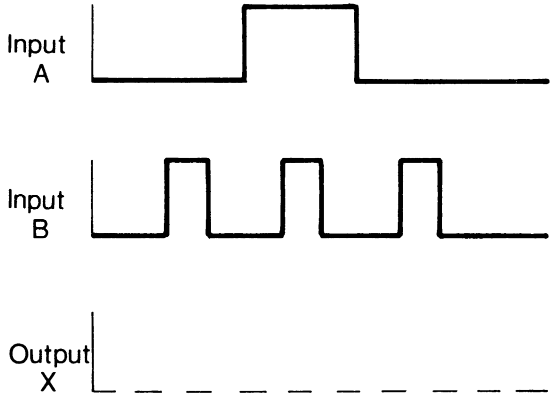


Fig. 1.9 Exercise. Draw in the output levels at X.
The solution is on page 27.



The NOT OR function

The *NOT OR* function is abbreviated to *NOR* and is explained by figure 1.11.

SWITCH ANALOGY

Refer to figure 1.11. There will be no output ('low' level) when switches A, B, or both are closed. With either switch closed the transistor is biased into saturation giving a 'low' output. If switches A and B are both open, VT3 will receive no bias. There will be an output ('high' level).

A SIMPLE EQUIVALENT CIRCUIT

A simple equivalent circuit is shown in figure 1.12. If A, B, or both, are 'high' level, the output voltage is 'low' level. VT1 or VT2 being on, will produce a voltage drop across R1. The voltage drop across R1 will bias VT3 into saturation producing a 'low' output. If both VT1 and VT2 are not conducting due to 'low' level inputs there will be no volts dropped across R1. VT3 will not be biased on, and the output level will be 'high'.

THE CIRCUIT SYMBOL

As the *NOR* gate is an *OR* function with an inverting transistor (VT3) after it, the circuit symbol could be drawn as shown in figure 1.10a.

The actual circuit symbol for the *NOR* gate is shown in figure 1.10b.

Fig. 1.10 The NOR gate circuit symbol.

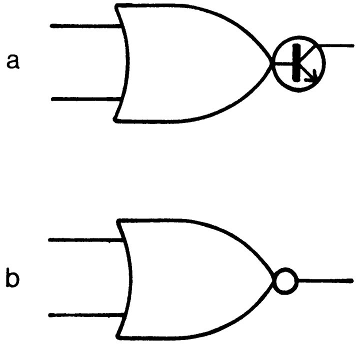


Fig. 1.11 Switches used as a NOR gate.

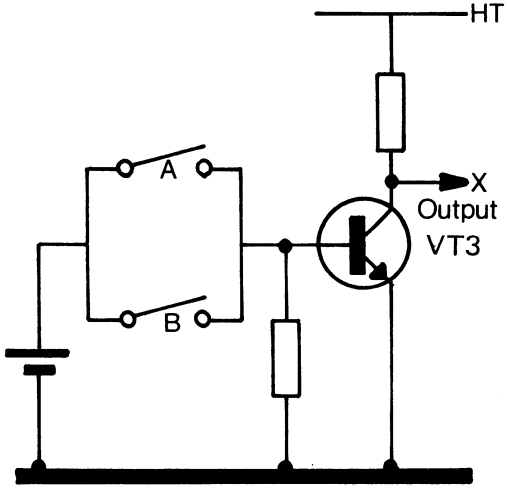
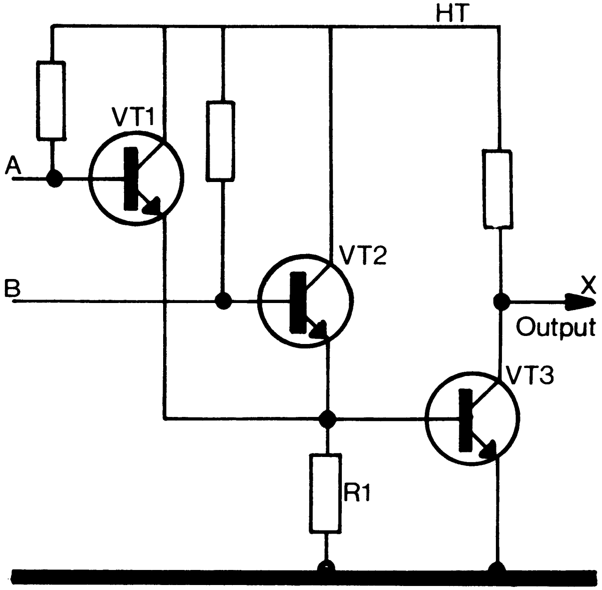


Fig. 1.12 The NOR gate from transistors.



VOLTAGE LEVEL TRUTH TABLE

A voltage level truth table for the NOR gate is shown in figure 1.13.

A PRACTICAL USE FOR THE NOR GATE

Switches X and Y are safety switches on an electrically driven machine. Only when the switches are both closed, will the *NOR* gate give a 'high' output to switch on T1 and operate the relay, which closes the motor switch, and starts the machine.

EXERCISE (FIGURE 1.16).

The illustration shows the input levels A and B of an *NOR* gate. Draw in the output levels at X. The solution is on page 27.

Fig. 1.13 The voltage level truth table.

Inputs		Output
A	B	X
H	H	L
H	L	L
L	H	L
L	L	H

Fig. 1.14 Summary of the truth table.

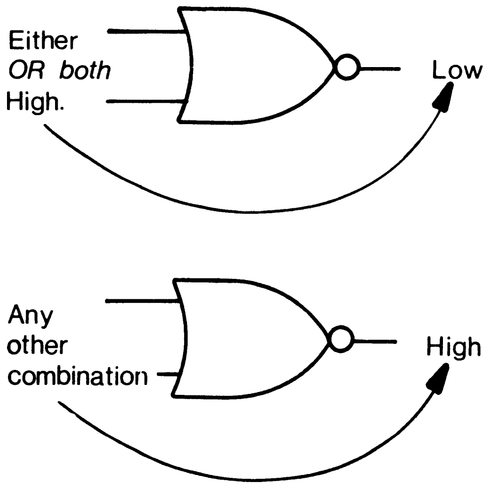


Fig. 1.15 A practical use for the NOR gate.

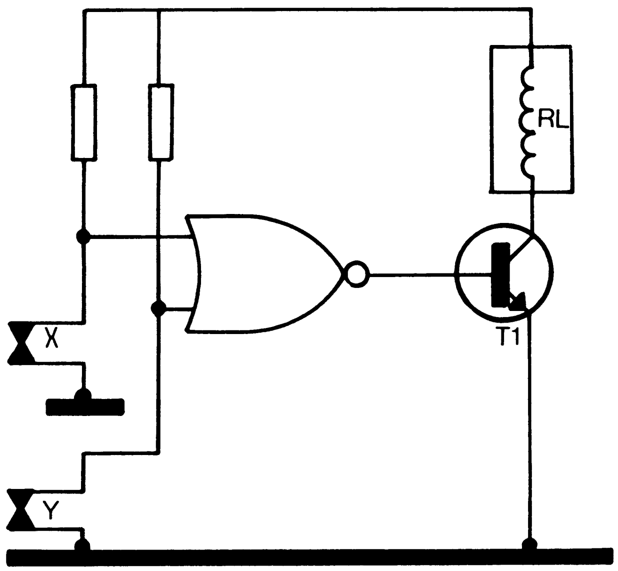
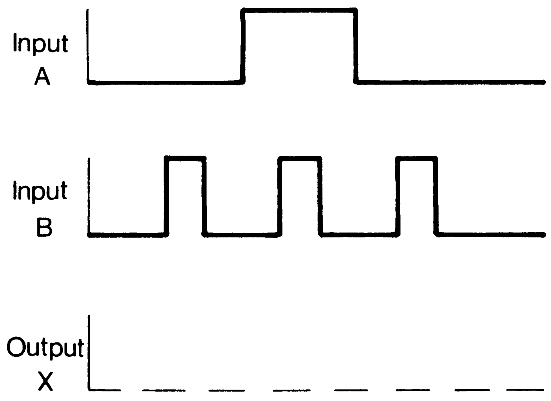


Fig. 1.16 Exercise. Draw in the output levels at X. The solution is on page 27.



The AND function

The *AND* function is explained by figure 1.18.

There will be an output voltage across R only when switch A *AND* switch B are closed.

If switches A, B, or both are open there will be no output voltage across R.

AN EQUIVALENT CIRCUIT

An *AND* gate using two NPN transistors is shown in figure 1.19.

If input A and input B are 'high' level, both transistors are switched on and the output voltage is 'high' level due to the voltage drop across R, since current is flowing through R.

If one, or both transistors are not conducting due to 'low' level inputs, there would be no volts drop across R and the output voltage level would be 'low'.

THE CIRCUIT SYMBOL

The symbol for the *AND* gate just described is shown in figure 1.17.

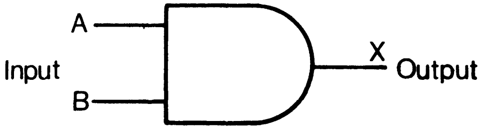


Fig. 1.17 The AND gate circuit symbol.

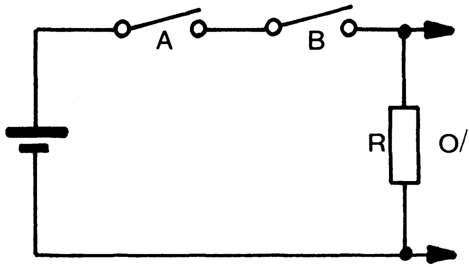


Fig. 1.18 Two switches as an AND gate.

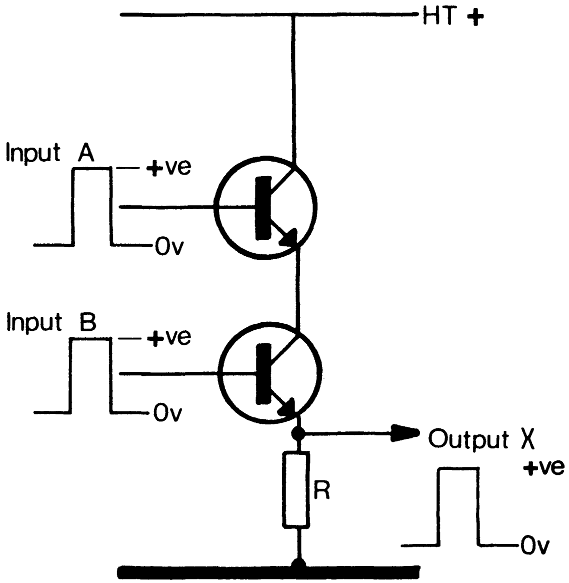


Fig. 1.19 An AND gate from two transistors.

VOLTAGE LEVEL TRUTH TABLE

A voltage level truth table for the *AND* gate is shown in figure 1.20.

A PRACTICAL USE OF THE AND GATE.

The diagram in figure 1.22 shows a practical use for the AND gate.

Switches X and Y are safety switches on an electrically driven machine. Only when switches X and Y are closed will the *AND* gate give a 'high' output to switch T1 and operate the relay, which closes the motor switch, and starts the machine.

EXERCISE (FIGURE 1.23)

The illustration shows the input levels A and B of an *AND* gate. Draw in the output levels at X. The solution is on page 27.

Fig. 1.20 The voltage level truth table.

Inputs		Output
A	B	X
H	H	H
H	L	L
L	H	L
L	L	L

Fig. 1.21 Summary of the truth table.

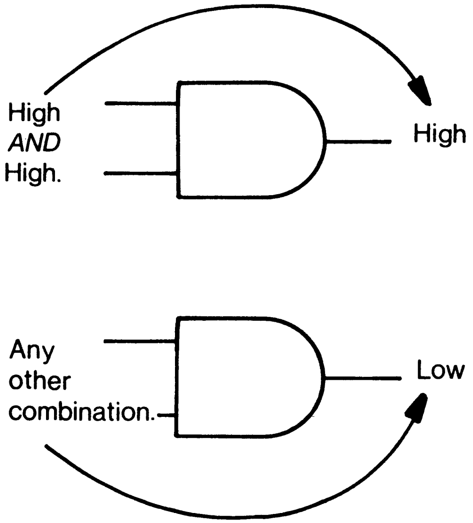


Fig. 1.22 A practical use for the AND gate.

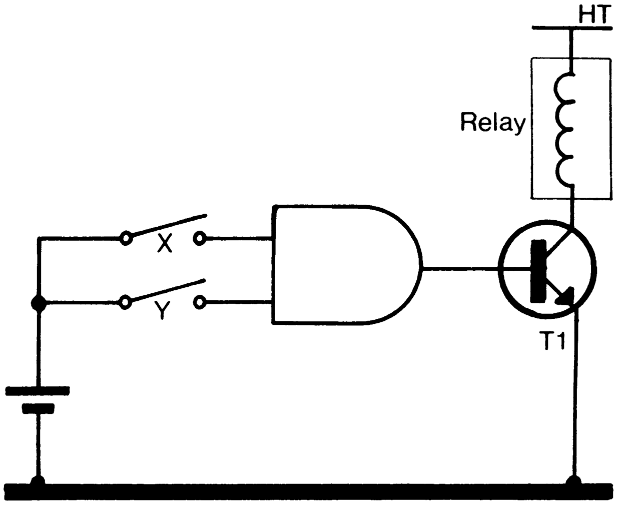
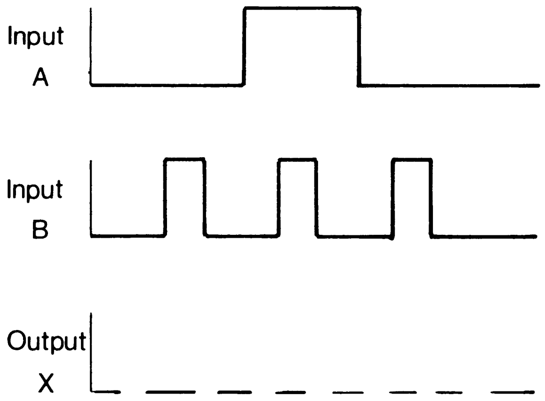


*Fig. 1.23 Exercise. Draw in the output levels at X.
The solution is on page 27.*



The NOT AND function

The *NOT AND* function is abbreviated to *NAND* function. It behaves in an opposite manner to the *AND* function and is explained by figure 1.25.

There will be no output ('low' level) when switches A and B are closed. With both switches closed VT1 is biased into saturation giving a 'low' output. If either, or both switches are open, VT3 receives no bias and there is an output ('high' level).

AN EQUIVALENT TRANSISTOR CIRCUIT.

An equivalent transistor circuit is shown in figure 1.26. If inputs A and B are 'high' level the output voltage is 'low' level. VT1 and VT2 are both on, giving a voltage drop across R1. The voltage drop across R1 biases VT3 into saturation producing a 'low' output. If either or both transistors are off, due to 'low' level inputs there will be no voltage drop across R1, VT3 will not be biased on, and the output level will be 'high'.

THE NAND GATE CIRCUIT SYMBOL.

As the *NAND* gate is an *AND* function with an inverting transistor (VT3) after it, the circuit symbol could be drawn as shown in figure 1.24a.

The circuit symbol for the *NAND* gate is shown in figure 1.24b.

Fig. 1.24 The NAND gate circuit symbol.

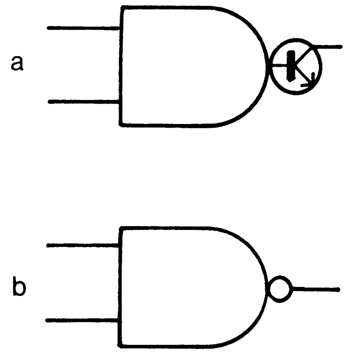


Fig. 1.25 Two switches as a NAND gate.

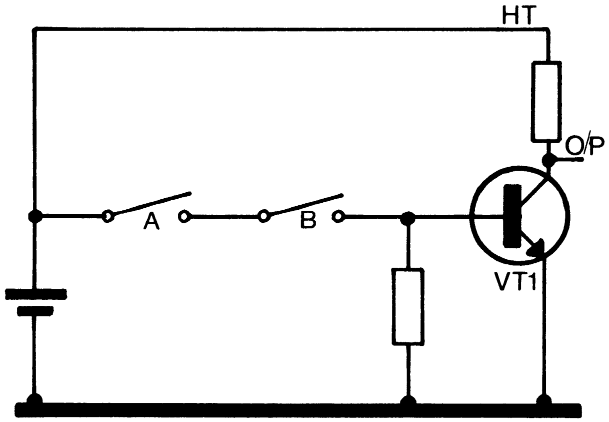
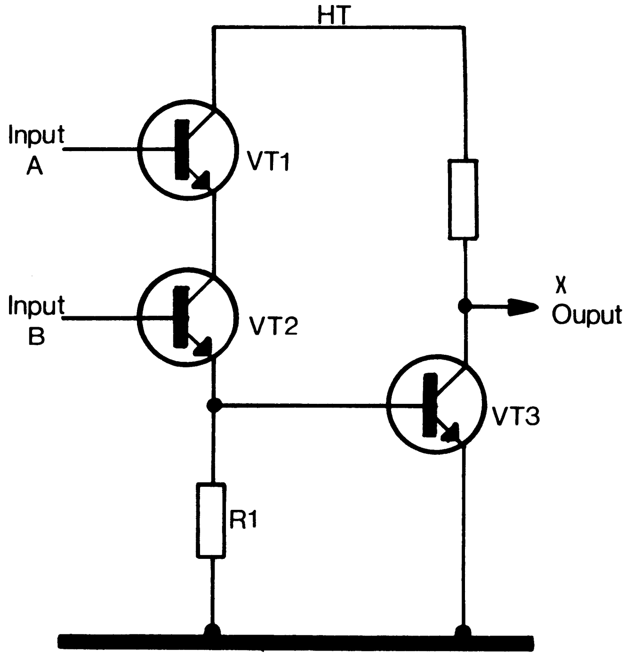


Fig. 1.26 An equivalent transistor circuit.



VOLTAGE LEVEL TRUTH TABLE.

A voltage level truth table is shown in figure 1.27.

A PRACTICAL USE OF THE NAND GATE.

Figure 1.29 details a practical use for a *NAND* gate. Input B only produces an output when input A is 'high'. Input A could act as an enable input to look at information at a specific time. The example is similar to a burst gate: The input cycles at B can only pass when the pulse at A is present. Any signal outside the pulse at A does not appear at the output.

EXERCISE (FIGURE 1.30)

The illustration shows the input levels A and B of a *NAND* gate. Draw in the output at X. The solution is on page 27.

Fig. 1.27 The voltage level truth table.

Inputs		Output
A	B	X
H	H	L
L	H	H
H	L	H
L	L	H

Fig. 1.28 Summary of the truth table.

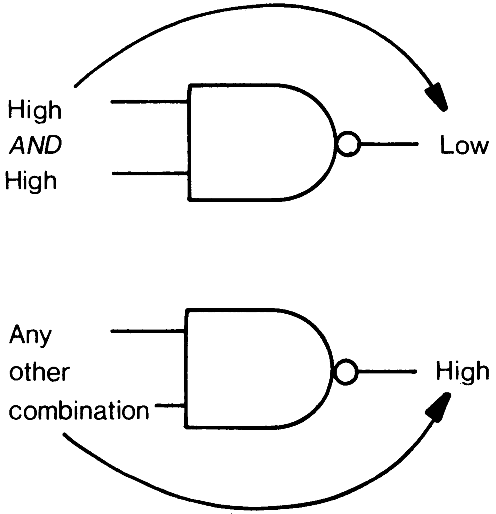


Fig. 1.29 A practical use of the NAND gate.

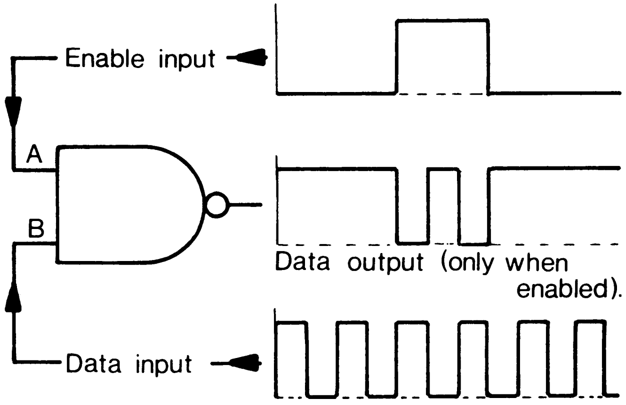
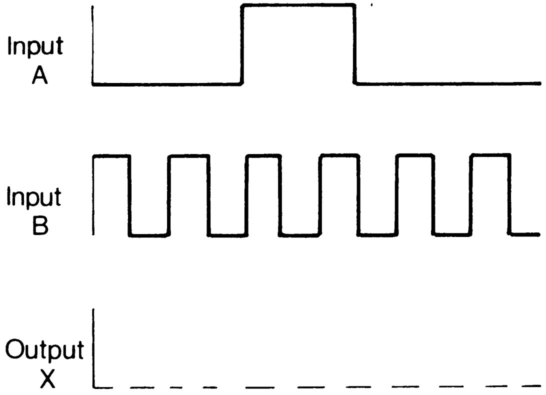


Fig. 1.30 Exercise. Draw in the output levels at X. The solution is on page 27.



The EXCLUSIVE OR function

The *EXCLUSIVE OR* gate is explained by figure 1.34. There will be an output across R when only one of switches A or B is closed. If both switches are open or; both switches are closed; then no output occurs.

The circuit symbol for an *EXCLUSIVE OR* gate is shown in figure 1.31 and the voltage level truth table is shown in figure 1.32.

EXERCISE (FIGURE 1.35).

The illustration shows the input levels A and B of an *EXCLUSIVE OR* gate. Draw in the output levels at X. The solution is on page 27.

Fig. 1.31 The EXCLUSIVE OR gate circuit symbol.

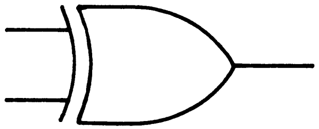


Fig. 1.32 The voltage level truth table.

Inputs		Output
A	B	X
L	L	L
L	H	H
H	L	H
H	H	L

Fig. 1.33 Summary of the truth table.

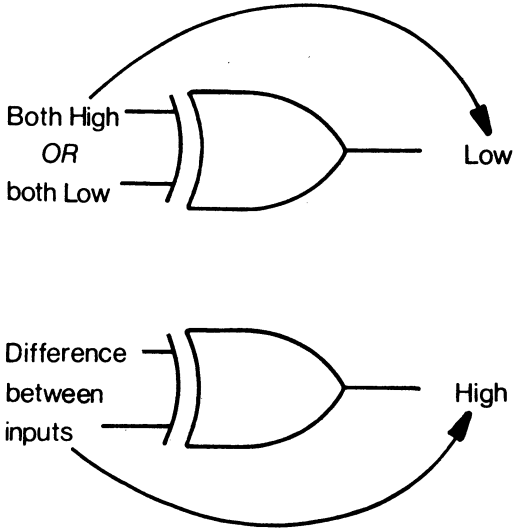


Fig. 1.34 An EXCLUSIVE OR gate from switches.

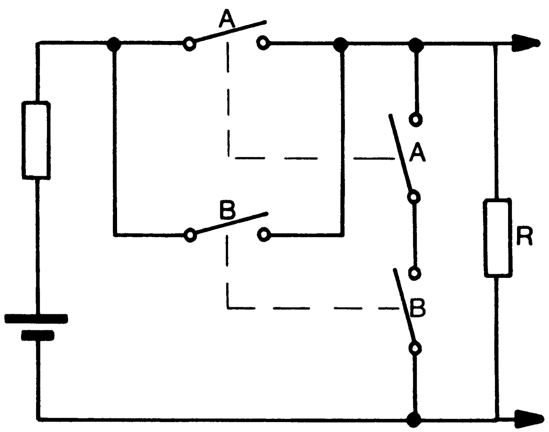
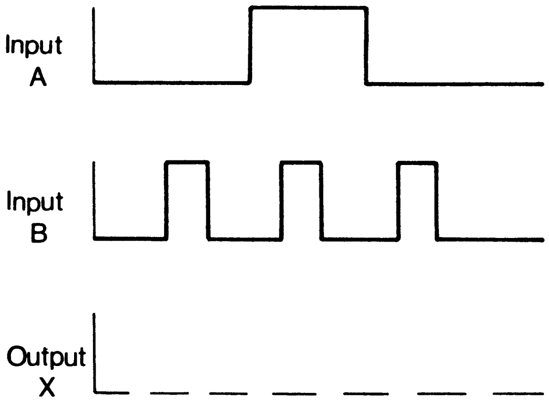


Fig. 1.35 Exercise. Draw in the output levels at X. The solution is on page 27.



THE EXCLUSIVE OR GATE PRACTICAL USE.

The schematic diagram (Figure 1.36), shows the circuit of a coincidence detector.

The *EXCLUSIVE OR* gate gives a 'low' output when both its' inputs are the same, and so may be used as a coincidence detector.

0° PHASE DIFFERENCE (FIGURE 1.37-1).

If in phase (coincident), the inputs are always the same and so a 'low' output is obtained.

180° PHASE DIFFERENCE (FIGURE 1.37-2).

If out of phase (not coincident), the inputs are always different and so a 'high' output is obtained.

90° PHASE DIFFERENCE (FIGURE 1.37-3).

The inputs are coincident and not coincident, and so the output changes between 'low' and 'high'. If the output is averaged the result would be a voltage of half the 'high' voltage.

CONTROL OF OSCILLATOR

Refer to figure 1.36. The oscillator output is compared with the reference signal: They are locked 90° apart. Therefore the situation is as shown in figure 1.37-3. i.e. an average voltage is produced. If the oscillator goes out of lock either way, then the output from the *EXCLUSIVE OR* gate will, via the smoothing circuit, produce a DC voltage to correct the oscillator.

Fig. 1.36 A coincidence detector.

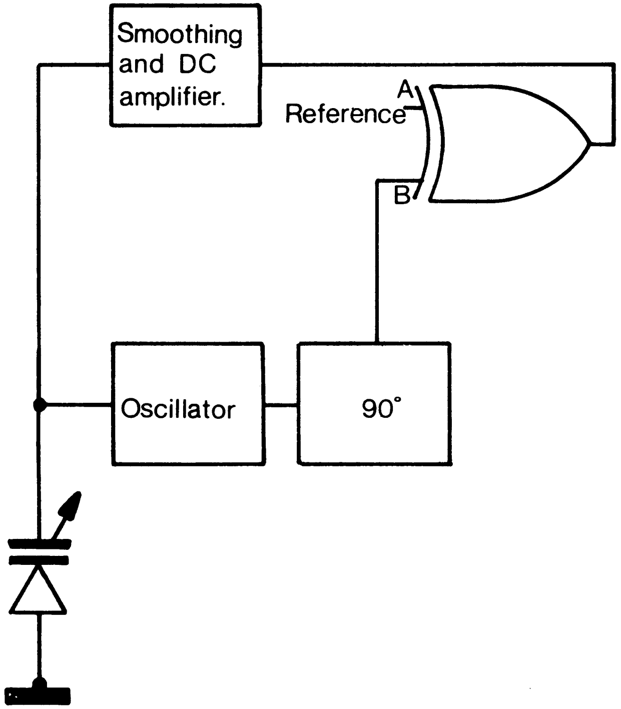


Fig. 1.37-1. 0° phase difference. (Oscillator lagging.)

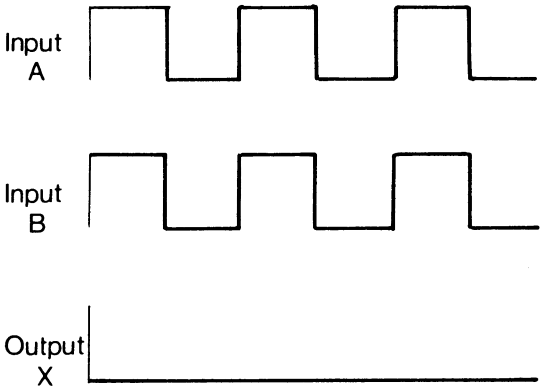


Fig. 1.37-2. 180° phase difference. (Oscillator leading.)

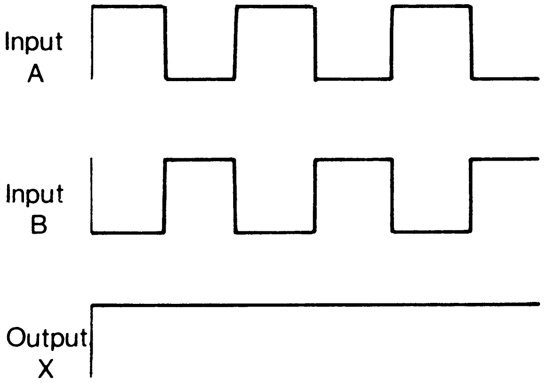
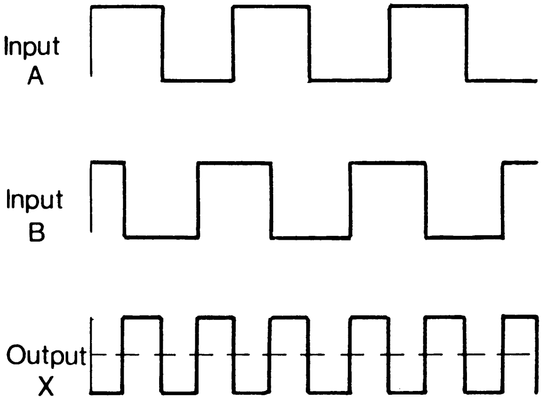


Fig. 1.37-3. 90° phase difference. (Oscillator correct.)



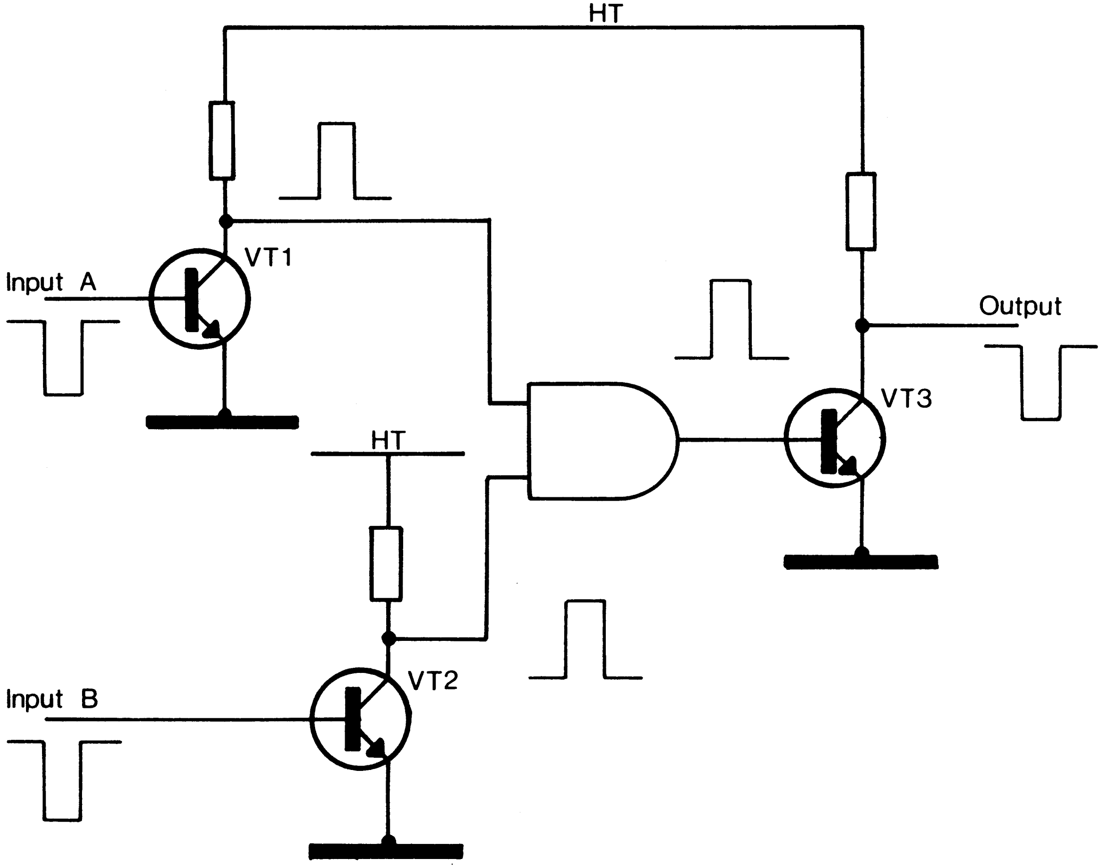
Gates operated by low level input pulses

So far, we have considered gates operated by 'high' level input pulses. Gates operated by 'low' input pulses will now be considered.

AND FUNCTION OPERATED BY LOW LEVEL INPUT PULSES.

Refer to figure 1.38. VT1 and VT2 invert the 'low' level input pulses and produce 'high' level pulses at their collectors. Two 'high' level pulses applied to an *AND* function will give a high level output. VT3 will invert the pulse and the output will be a 'low' level pulse.

Fig. 1.38 The AND function with inverting transistors.



THE CIRCUIT SYMBOL.

As this gate is an *AND* gate with inverting transistors on its inputs, and an inverting transistor on its output, the circuit symbol could be drawn as shown in figure 1.39.

The actual circuit symbol used is shown in figure 1.40.

VOLTAGE LEVEL TRUTH TABLE AND SUMMARY.

The voltage level truth table for this gate is shown in figure 1.41. A summary of its function is given in figure 1.42.

Fig. 1.39 A possible circuit symbol.

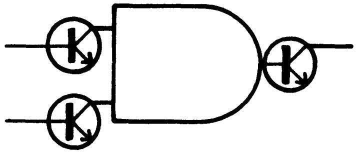


Fig. 1.40 The circuit symbol used.

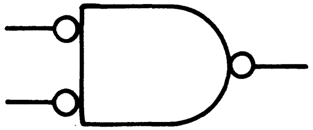
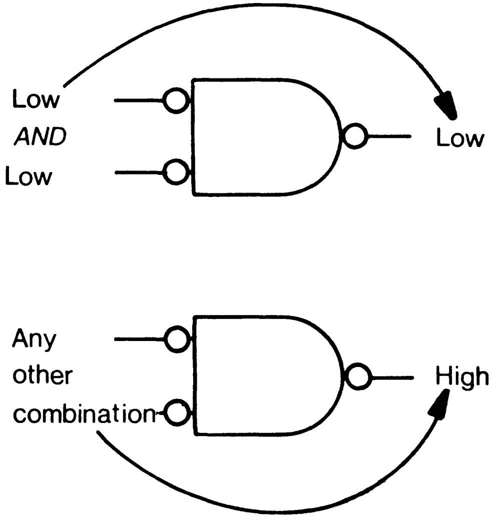


Fig. 1.41 Voltage level truth table.

Inputs		Output
A	B	X
L	L	L
H	L	H
L	H	H
H	H	H

Fig. 1.42 Summary of operation.

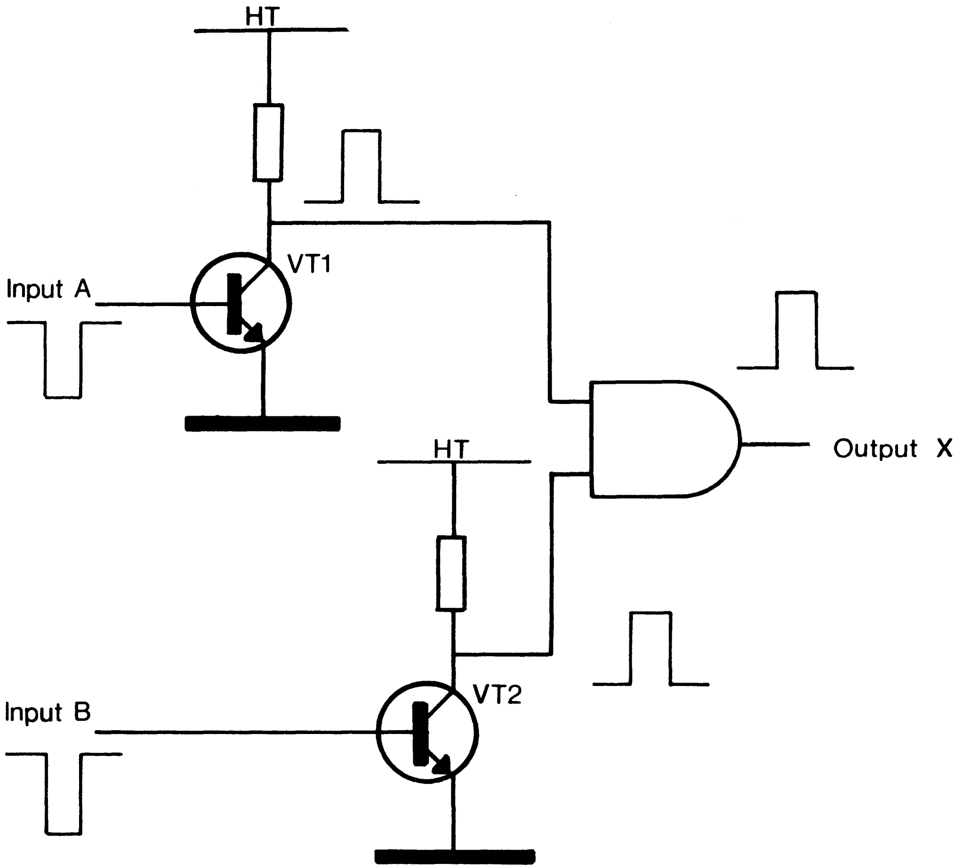


THE NAND FUNCTION OPERATED BY LOW LEVEL INPUT PULSES.

Refer to figure 1.43, VT1 and VT2 invert the 'low' level input pulses and produce 'high' level pulses at their collectors.

Two 'high' level pulses applied to an *AND* function will give a 'high' level output

Fig. 1.43 The AND function with inverting transistors.



THE CIRCUIT SYMBOL.

As this gate is an *AND* gate with inverting transistors on its inputs, the circuit symbol could be drawn as shown in figure 1.44. The actual circuit symbol used is shown in figure 1.45.

VOLTAGE LEVEL TRUTH TABLE AND SUMMARY.

The voltage level truth table for this gate is shown in figure 1.46. A summary of its function is given in figure 1.47.

Fig. 1.44 A possible circuit symbol.

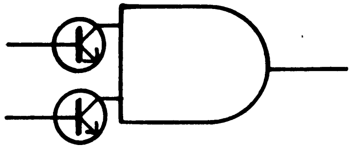


Fig. 1.45 The circuit symbol used.

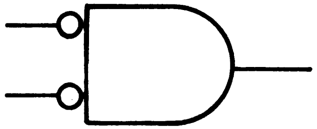
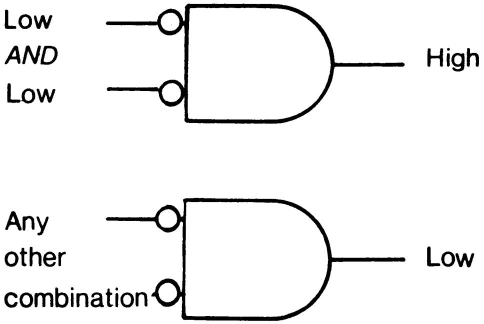


Fig. 1.46 The voltage level truth table.

Inputs		Output
A	B	X
H	H	L
H	L	L
L	H	L
L	L	H

Fig. 1.47 Summary of operation.



THE OR FUNCTION OPERATED WITH LOW LEVEL INPUTS.

The *OR* function with inverting transistors on its inputs and output could be drawn as shown in figure 1.48.

The actual circuit symbol used is as shown in figure 1.49.

THE VOLTAGE LEVEL TRUTH TABLE AND SUMMARY OF OPERATION.

The voltage level truth table for this gate is shown in figure 1.50. A summary of its function is given in figure 1.51.

Fig. 1.48 A possible circuit symbol.

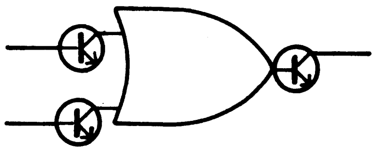


Fig. 1.49 The circuit symbol used.

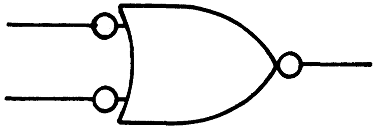
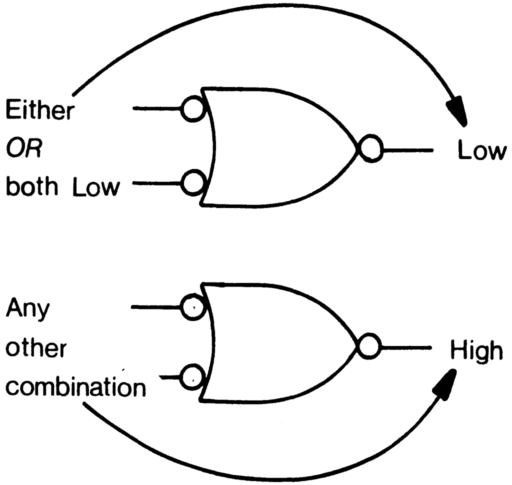


Fig. 1.50 The voltage level truth table.

Inputs		Output
A	B	X
H	H	H
H	L	L
L	H	L
L	L	L

Fig. 1.51 Summary of operation.



THE NOR FUNCTION OPERATED WITH LOW LEVEL INPUTS.

The *NOR* function with inverting transistors on its inputs could be drawn as shown in figure 1.52.

The actual circuit symbol used is as shown in figure 1.53.

THE VOLTAGE LEVEL TRUTH TABLE AND SUMMARY OF OPERATION.

The voltage level truth table for this gate is shown in figure 1.54. A summary of its function is given in figure 1.55.

Fig. 1.52 A possible circuit symbol.

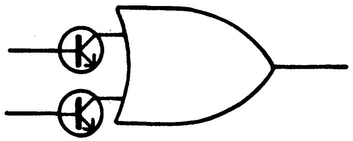


Fig. 1.53 The circuit symbol used.

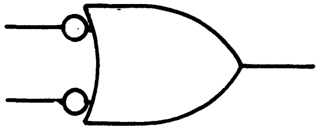
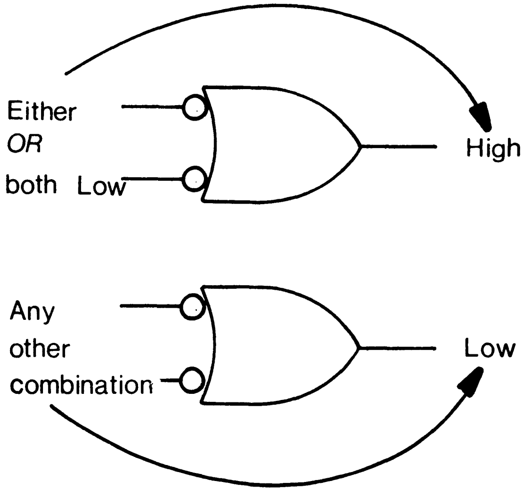


Fig. 1.54 The voltage level truth table.

Inputs		Output
A	B	X
H	H	L
H	L	H
L	H	H
L	L	H

Fig. 1.55 Summary of operation.



The BUFFER AMPLIFIER function

The buffer amplifier provides isolation between stages. It is used to prevent 'loading' effects. It can be used to increase the number of loads that the output from a gate feeds.

The circuit symbol of a buffer amplifier is shown in figure 1.56

The illustration figure 1.57 summarises the operating voltage levels of the buffer amplifier.

A PRACTICAL EXAMPLE.

Refer to figure 1.58. The output from a gate can feed up to ten loads. (A load is a single input to another gate).

Using a buffer amplifier, the output 'X' can feed nine loads. Therefore instead of only ten, the gate can feed a total of nineteen loads.

Fig. 1.56 The circuit symbol of a buffer amplifier.

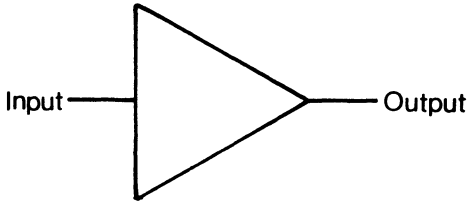


Fig. 1.57 Summary of operation.

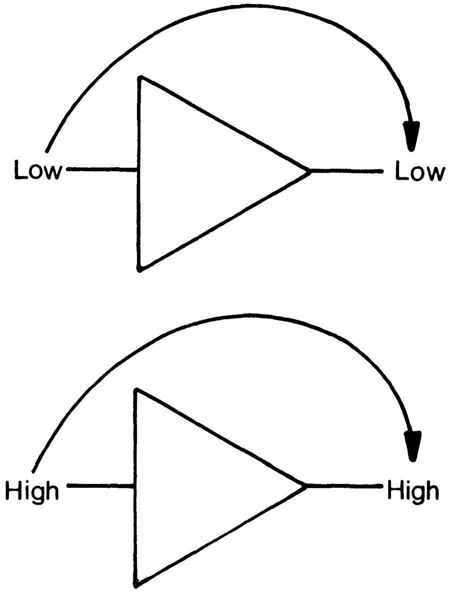
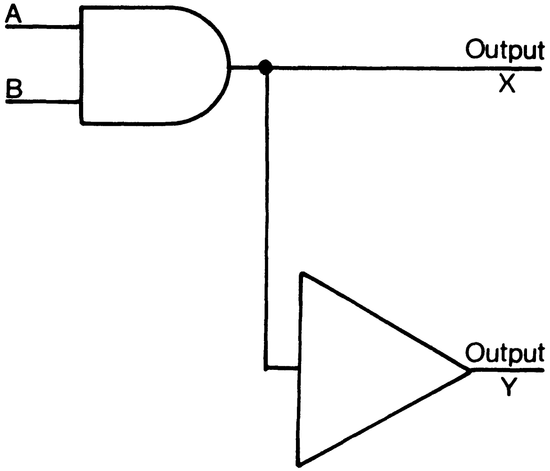


Fig. 1.58 Practical use for a buffer amplifier.



The INVERTING AMPLIFIER function

The inverting amplifier changes the logic state from 'low' to 'high' and vice versa. It can be used to provide this function or to change the characteristics of other gates.

The circuit symbol of the inverting amplifier is shown in figure 1.59 and a summary of its operating voltage levels is given in figure 1.60.

A PRACTICAL EXAMPLE.

Refer to figure 1.61. The inverting amplifier may be used to invert the input signal to a gate. This inverts the signal to the correct state to operate the gate. As shown in figure 1.62, the inverting amplifier can also be used to invert the output from a gate.

Fig. 1.59 The circuit symbol of an inverting amplifier.

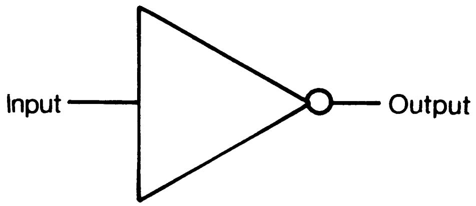


Fig. 1.60 Summary of operation.

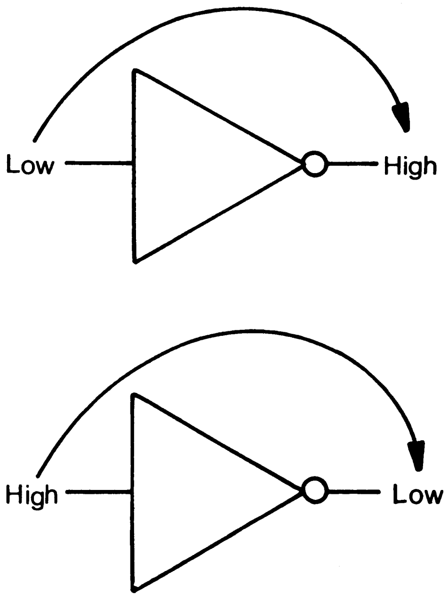


Fig. 1.61 Using the amplifier to invert an input signal.

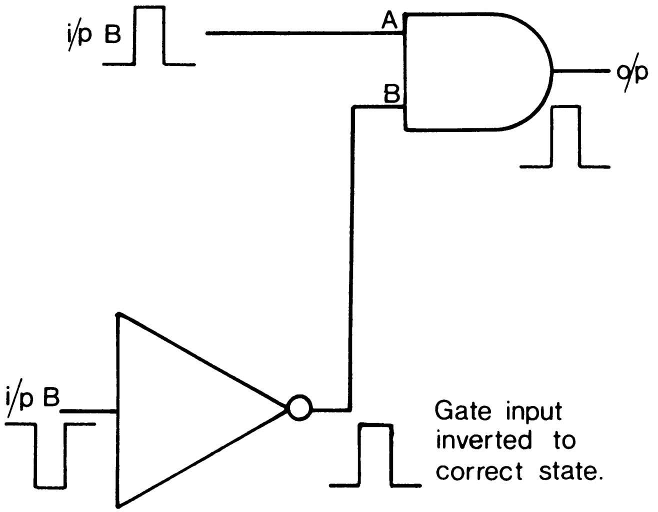
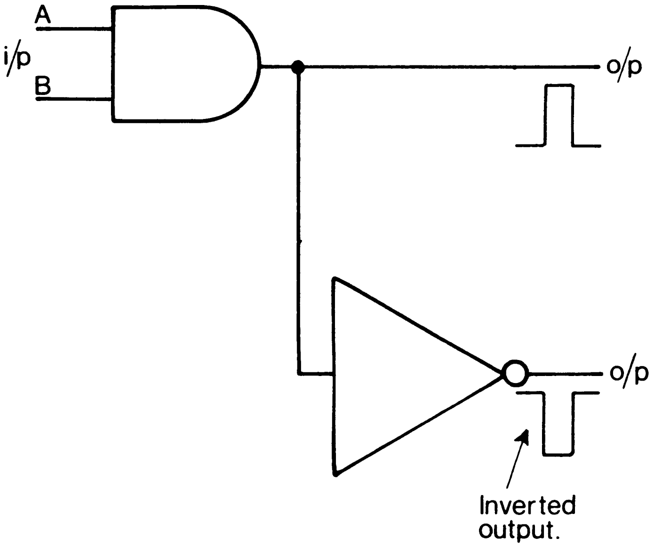


Fig. 1.62 Using the amplifier to invert an output signal.



A NOT gate from other gates

EXPLANATION.

Inverters are available in separate or mixed packages but, for economy reasons, an inverter function is sometimes made using *NOR* or *NAND* gates.

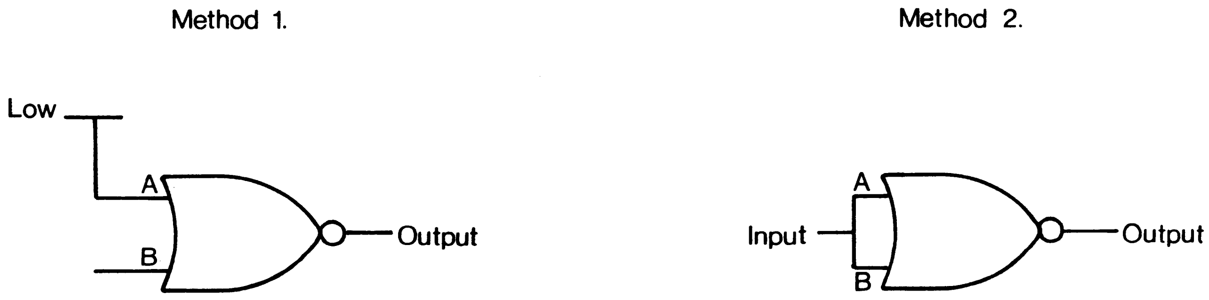
USING A NOR GATE.

With a *NOR* gate no output will occur when either of its inputs are high.

Method 1. Figure 1.63. Input A is tied to 'low'. If input B now goes to 'high' the output goes 'low' and vice versa.

Method 2. Figure 1.63. A and B are tied together. With a 'high' input the output goes 'low' and vice versa.

Fig. 1.63 Using a NOR gate as a NOT gate.



Method 1.

Method 2.

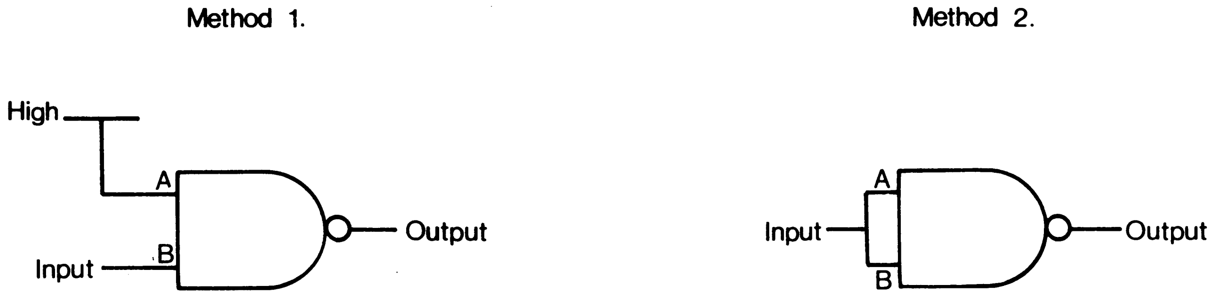
USING A NAND GATE.

With a *NAND* gate, no output will occur when inputs A and B are 'high'.

Method 1. Figure 1.64. Input A is tied to 'high'. If input B now goes 'high' the output will go 'low' and vice versa.

Method 2. Figure 1.64. Inputs A and B are tied together. With a 'high' input there will be a 'low' output and vice versa.

Fig. 1.64 Using a NAND gate as a NOT gate.



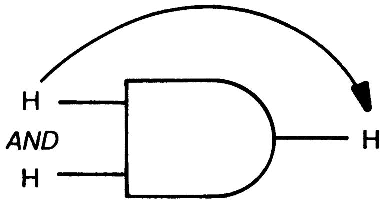
Method 1.

Method 2.

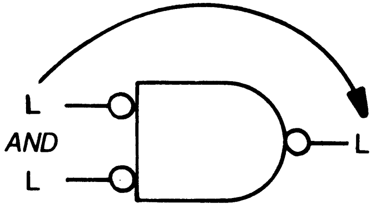
Summary of gate symbols

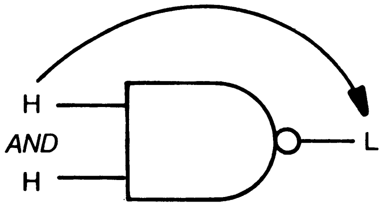
Gates that are operated by high level pulses are illustrated across the top of these two pages.

Gates operated by low level pulses are across the bottom of these two pages.

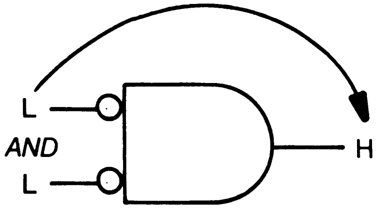


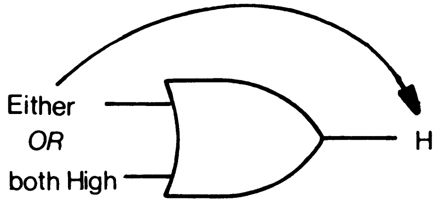
AND



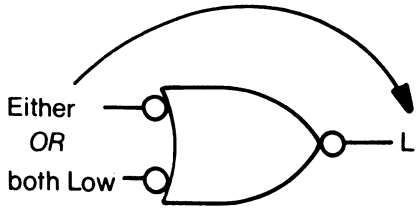


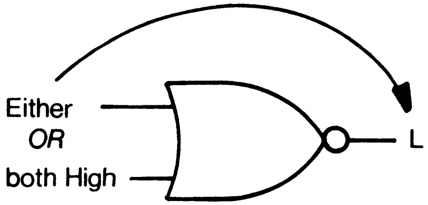
NAND



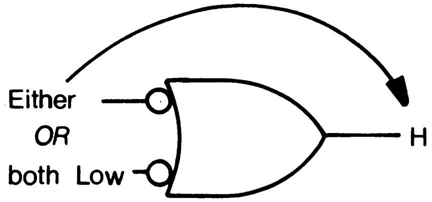


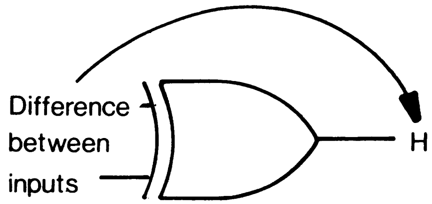
OR



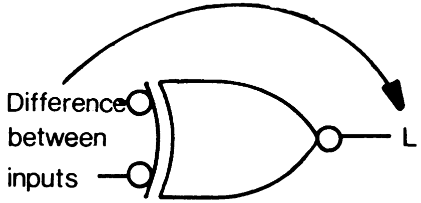


NOR





EXCLUSIVE OR



Revision

LOGIC SYMBOLS.

As we have seen, there are two basic types of symbols used. Figure 1.65 shows the shape of the symbol for the *OR* function. Figure 1.66 shows the shape of the symbol for the *AND* function.

STATE INDICATOR CIRCLE.

The circle on the input of the gate symbol is used to indicate the following conditions:—

CIRCLE ON THE INPUT (FIGURE 1.67).

The gate is operated by a relatively low input signal.

CIRCLE ON THE OUTPUT (FIGURE 1.68).

The output goes low when the gate is operated.

FIGURE 1.69.

The absence of a circle on the input, means that the gate is operated by a high level input signal.

The absence of a circle on the output, means that the output goes high when the gate is operated.

Fig. 1.65 The OR gate.

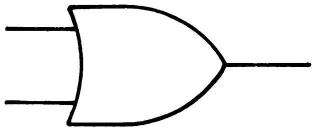


Fig. 1.66 The AND gate.

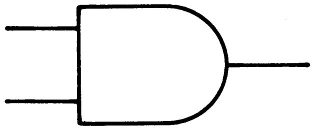


Fig. 1.67 Operated by a low input signal.

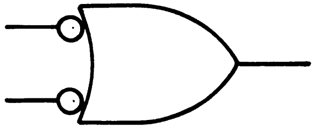


Fig. 1.68 Output goes low when operated.

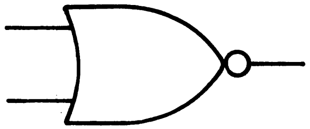
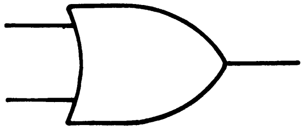
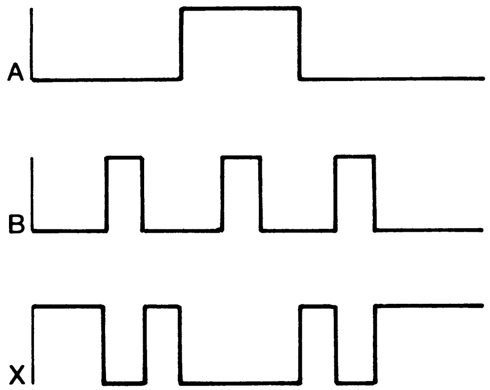


Fig. 1.69 The OR symbol.

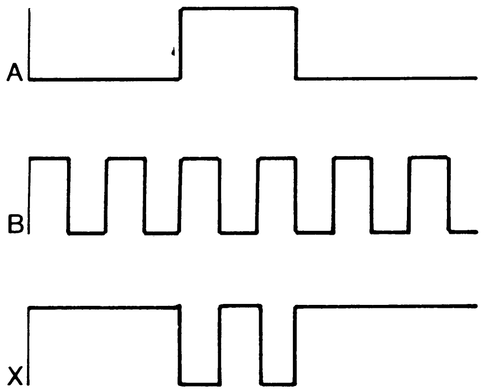


Solutions to exercises

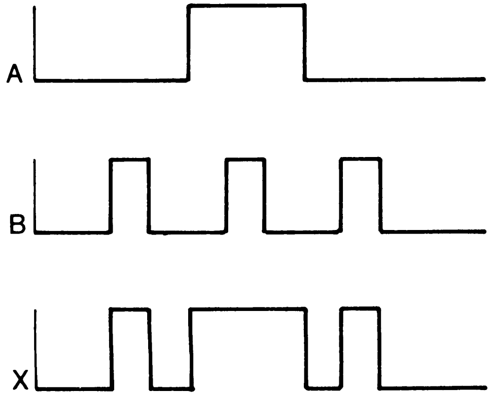
Page 7. Figure 1.16.



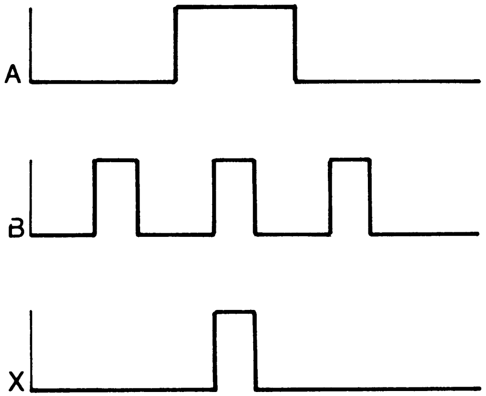
Page 11. Figure 1.30.



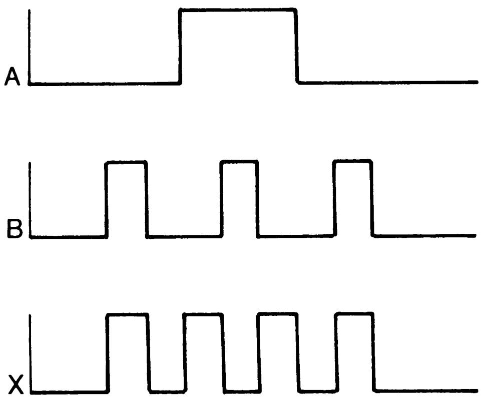
Page 5. Figure 1.9



Page 9. Figure 1.23



Page 12. Figure 1.35



Exercises

The following pages contain exercises designed to enable you to become familiar with using the logic gates described in this book. In each case it is a matter of breaking the circuit down into small sections and filling in the truth table. The solutions are given on pages 100–1.

Referring to figure 1.70, point C is the standard 'OR' gate output. Point D is the standard 'NAND' gate output. Output X has inputs of C and D and will *NAND* the inputs.

Fill in the output column. What gate could these three gates replace?

For economy or availability it is possible to make up all gates from just one type of gate, i.e. *NAND*.

Complete the truth tables for the following combinations and work out the type of gate they could replace.

Fig. 1.70 Exercise 1.

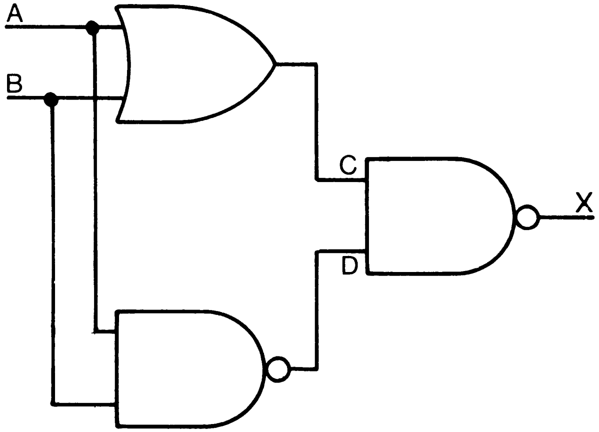


Fig. 1.71 The truth table.

Inputs				Output
A	B	C	D	
L	L	L	H	
L	H	H	H	
H	L	H	H	
H	H	H	L	

Fig. 1.72 *Exercise 2.*

Input		Outputs	
A	B	C	X
L	L		
L	H		
H	L		
H	H		

________ gate

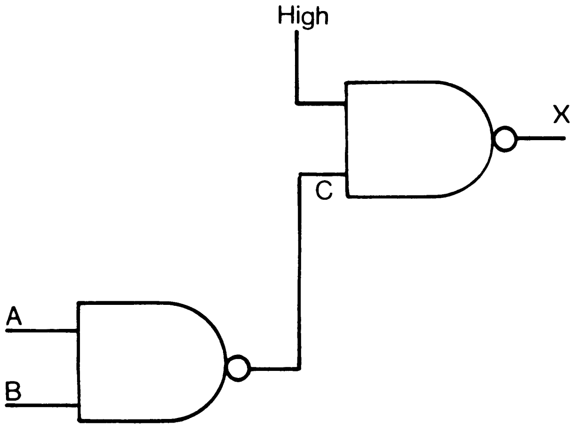


Fig. 1.73 *Exercise 3.*

Input		Outputs			
A	B	C	D	E	X
L	L				
L	H				
H	L				
H	H				

________ gate

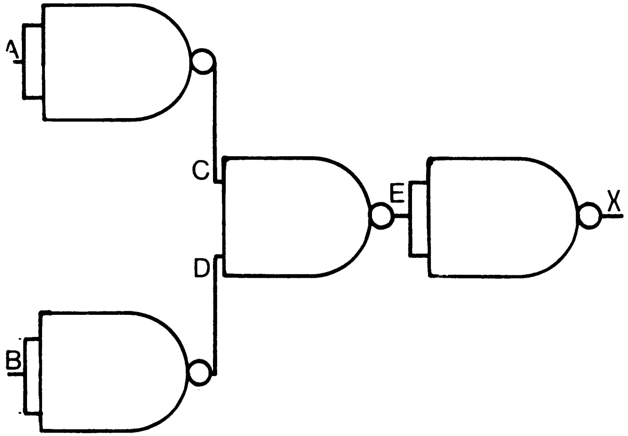


Fig. 1.74 *Exercise 4.*

Input		Outputs			
A	B	C	D	E	X
L	L				
L	H				
H	L				
H	H				

________ gate

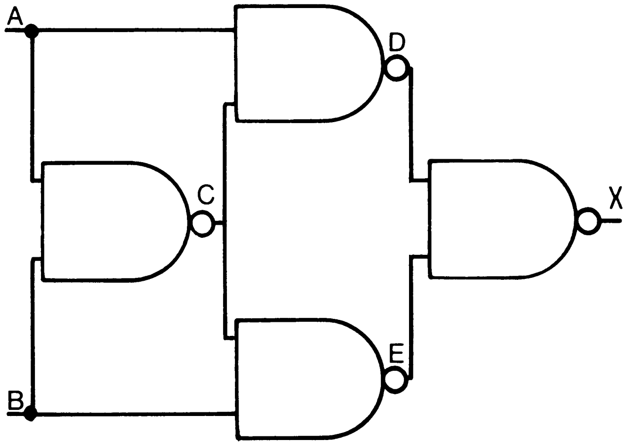
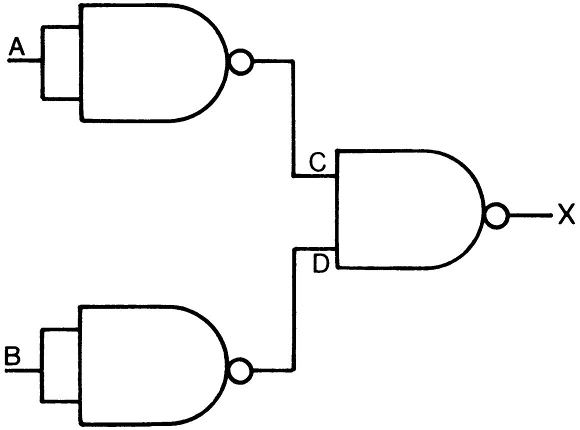


Fig. 1.75 Exercise 5.

Inputs		Outputs		
A	B	C	D	X
L	L			
L	H			
H	L			
H	H			

______ gate



USING NOR GATES.

It is also possible to make up all gates using *'NOR'* packages.

Complete the truth tables for the following combinations and work out the type of gate they could replace.

Fig. 1.76 Exercise 6.

Inputs		Outputs		
A	B	C	D	X
L	L			
L	H			
H	L			
H	H			

______ gate

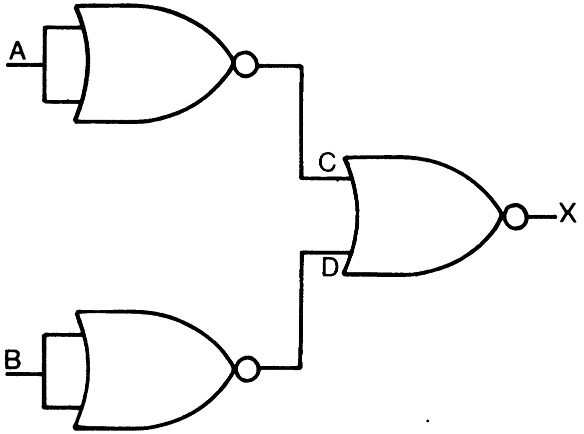


Fig. 1.77 Exercise 7.

Inputs		Outputs			
A	B	C	D	E	X
L	L				
L	H				
H	L				
H	H				

_____ gate

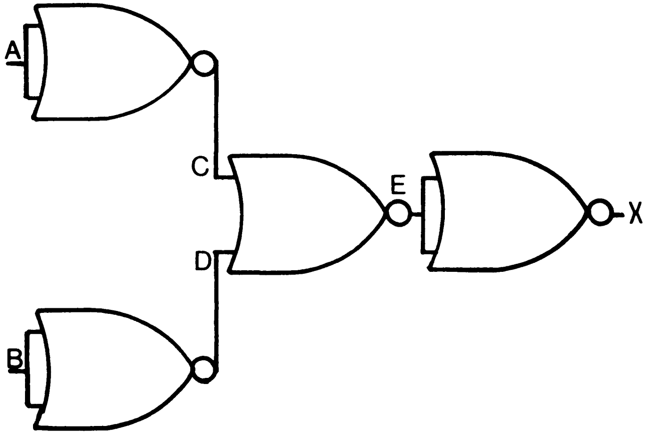
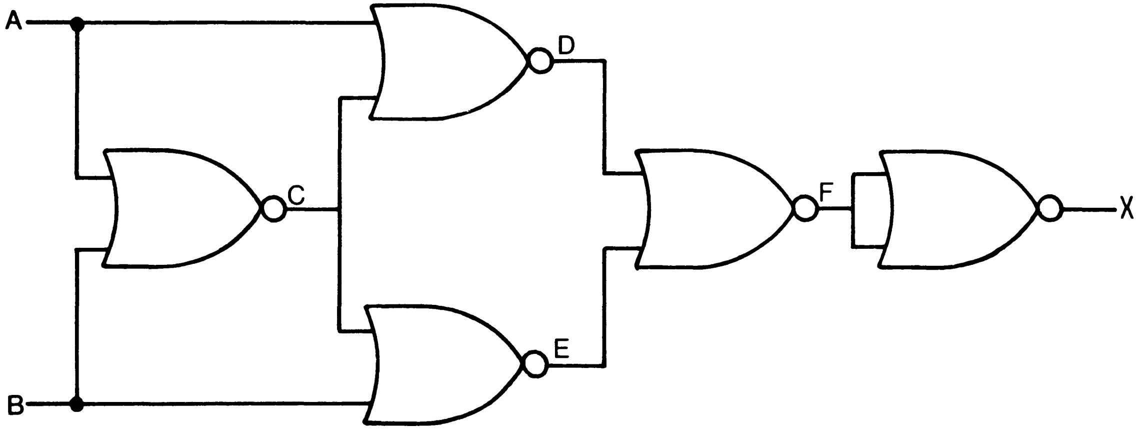


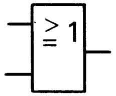
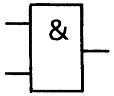
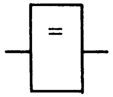
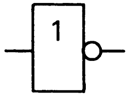
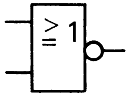
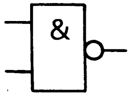
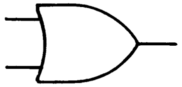
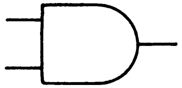
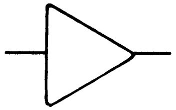
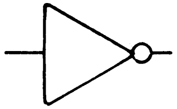
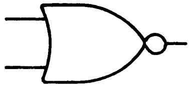
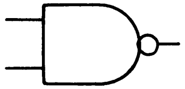
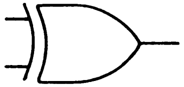
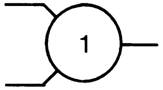
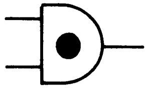
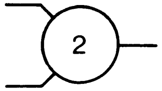
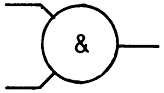
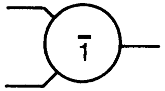
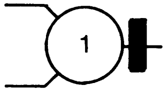
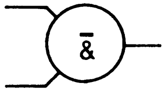
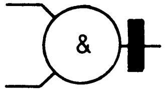
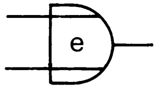
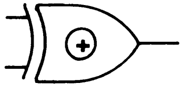
Fig. 1.78 Exercise 8.

Inputs		Outputs				
A	B	C	D	E	F	X
L	L					
L	H					
H	L					
H	H					

Exclusive __ gate



A synopsis of gate symbols

Gate type ➨	OR	AND	BUFFER	NOT	NOR	NAND	EXCLUSIVE 'OR'
British symbols							
American symbols							
Other symbols used		  			 	 	 
Inputs A B	Output X	Output X	Output X	Output X	Output X	Output X	Output X
L L	L	L	L	H	H	H	L
L H	H	L			L	H	H
H L	H	L			L	H	H
H H	H	H	H	L	L	L	L

Part 2

Flip-flops

INTRODUCTION.

Part 2 introduces the basics of *FLIP-FLOPS* and provides a foundation for later books in this series. The *FLIP-FLOP* is not a new device; it has appeared in television receivers since the introduction of colour. In colour television it appears in numerous decoders where it is called a bistable.

The transistor R-S FLIP-FLOP

In figure 2.1, VT2 has a low value base feed resistor (R1) giving a high base bias voltage (0.7v). VT2 is switched hard on with its collector voltage at approximately 0.2v.

Refer to figure 2.2. If the collector voltage of VT2 were used to apply bias to another transistor (VT1), it would be of an insufficient level to turn it on. To convert the circuit to a *FLIP-FLOP,* the resistor R1 is connected to VT1 collector. As this is at HT at present it will not affect the bias of VT2. This is shown in figure 2.2.

The circuit shown in figure 2.3 is a transistor *R-S FLIP-FLOP.* It is in a stable state which will continue until made to change by an external factor. To change state, the base of a transistor is taken 'low', e.g. shorted to chassis.

In figure 2.3, if VT2 base is taken 'low', VT2 will go off and its collector voltage will rise to HT and, via R2, VT1 will be biased on. VT1 collector will go'low'. If the short is now removed, VT2 will not come on again as, via R1, it is returned to a 'low' state of about 0.2v. In the new state VT1 collector is 'low' and VT2 collector is 'high'. This new state will remain, or be remembered, until the other input is taken 'low'. The *FLIP-FLOP* can therefore remember one of two states.

Fig. 2.1 Low value R1 gives a high base voltage.

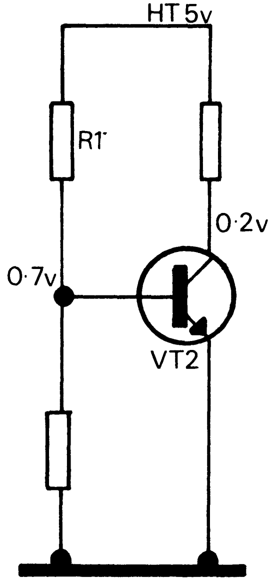


Fig. 2.2 VT1 cannot turn on; it has insufficient bias.

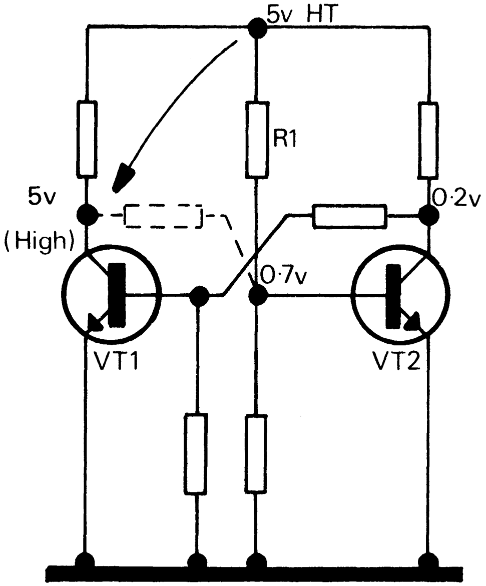
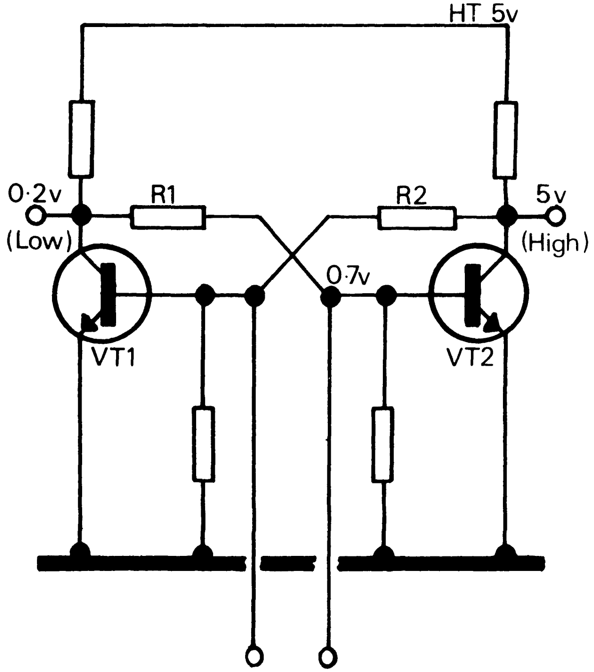


Fig. 2.3 The transistor R-S FLIP-FLOP.



THE TRANSISTOR R-S FLIP-FLOP (FIGURE 2.4)

The outputs taken from the transistor collectors are called Q and Q̄. The Q̄ output is the inverse of the Q output.

The inputs that are used to change the state of the *FLIP-FLOP* are called 'Set' and 'Reset' inputs.

FIGURE 2.5

If the Set input is taken 'low', e.g., shorted to chassis, bias is removed from VT2, and the Q output is set 'high'. The *FLIP-FLOP* will stay in this state even if the short circuit is now removed and the Set input is allowed to float.

FIGURE 2.6

If the Reset input is taken 'low', e.g., shorted to chassis, bias is removed from VT1 and Q̄ goes'high'; this biases VT2 on, resetting Q 'low'. The *FLIP-FLOP* will remain reset even if the short circuit is now removed and the Reset input is allowed to float.

If as shown in figure 2.7, both Set and Reset are taken 'low', bias is removed from VT1 and VT2, Q and Q̄ will be 'high'. If the short circuits are removed from Set and Reset at the same time, the bias to both transistors will rise. One of the transistors will start to conduct,which will make its collector voltage fall. The fall in voltage will be fed to the other transistor and will lower its bias, holding it off.

It is not possible to determine which transistor will conduct first and so the two 'low inputs state' is known as the 'undetermined' state.

Fig. 2.4 The transistor R-S FLIP-FLOP showing: the Set and Reset inputs, the Q and Q̄ outputs.

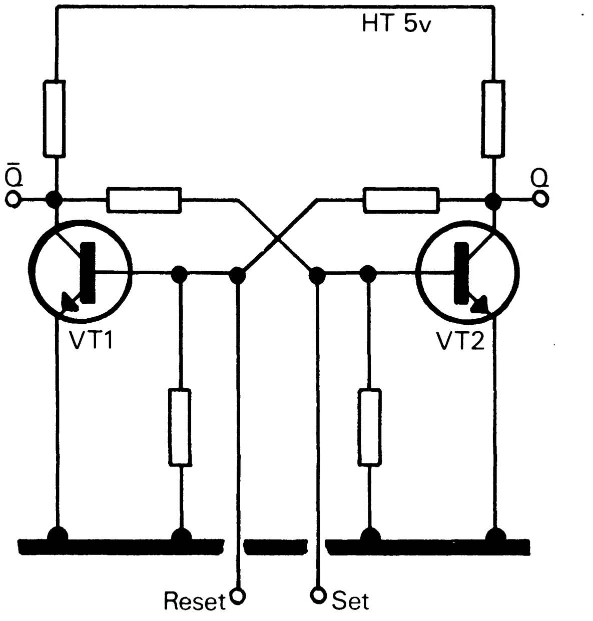


Fig. 2.5 The Set input shorted to chassis (taken low).

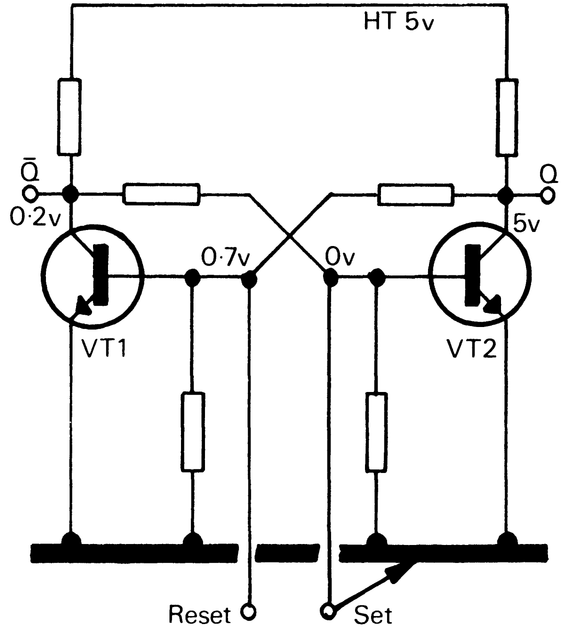


Fig. 2.6 The Reset input shorted to chassis (taken low).

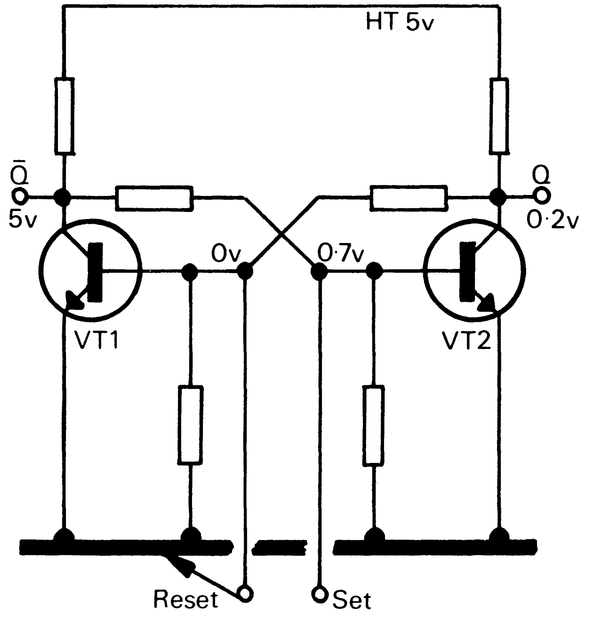
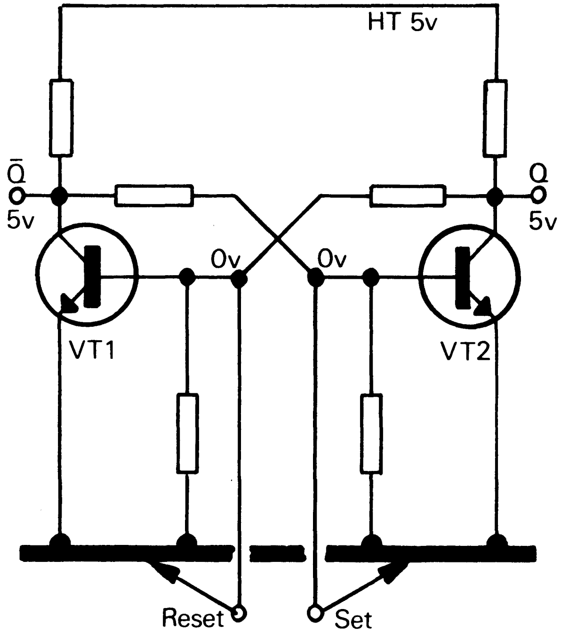


Fig. 2.7 Both Set and Reset inputs taken low.



PRACTICAL TRANSISTOR R-S FLIP-FLOP.

The *FLIP-FLOP* just described is used in the U704 and U705 remote controls (Figure 2.8), where it is known as 'the sound mute bistable'. Because R52 is of a larger value than R53, VT44 base is more positive than VT45 base at switch on. The circuit will always start with VT44 on and VT45 off.

The bistable works from one input and components are added to route the input to VT44 and VT45 alternately. This is achieved by W42, W43, R88 and R55.

FIGURE 2.9

If VT45 is off, its collector is 'high' which, fed via R55, reverse biases W43. VT44 is on, as it is biased from VT45 collector. VT44 collector is at 0.2v and its base is at 0.6 to 0.7 volts; therefore W42 is on the threshhold of conduction. If the input drops in a negative direction it is routed via W42 to switch VT44 off.

FIGURE 2.10

VT44 goes off, its collector rises and, via R49, biases VT45 on. VT45 collector falls and so has insufficient voltage on its collector to bias VT44 on. The bistable has changed state: VT44 is off, VT45 is on.

W42 is now reverse biased and W43 is forward biased. The next pulse will therefore pass via W43 to switch VT45 off.

The bistable changes state with each input pulse. This is known as toggling.

Fig. 2.8 The 'U704/5 sound mute bistable'.

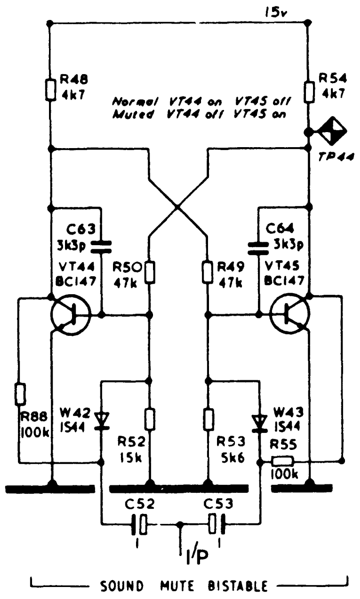


Fig. 2.9 VT45 off and VT44 on.

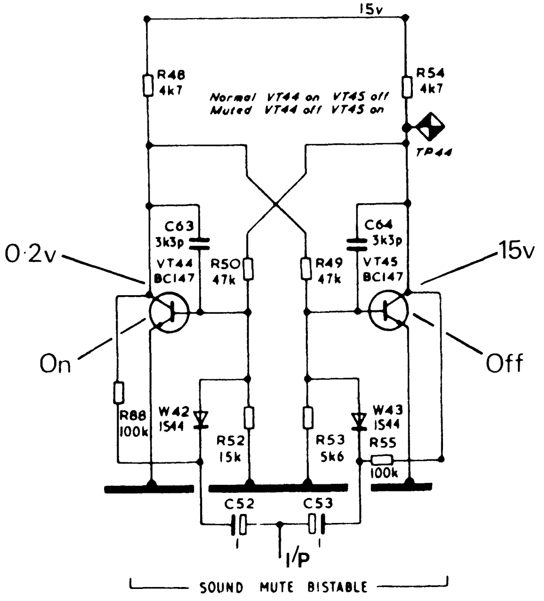
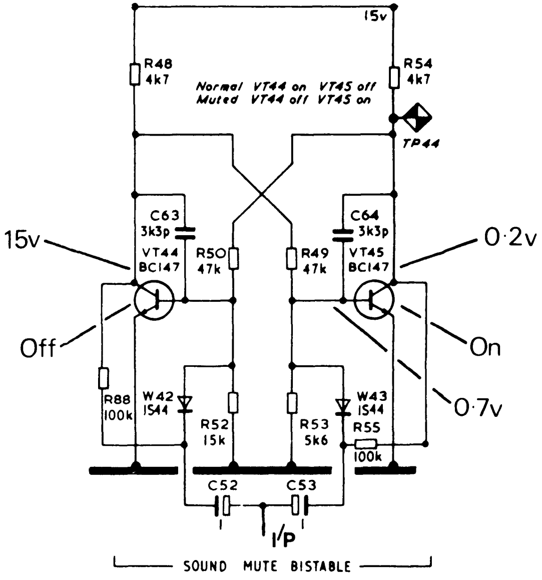


Fig. 2.10 VT44 off and VT45 on.



R-S FLIP-FLOPS from logic gates

There are several types of *FLIP-FLOPS,* some more versatile than others, but all are constructed from the basic *RESET-SET FLIP-FLOP.*

Low Input Activated

Figure 2.11 shows the circuit symbol of a 'Low input activated *R-S FLIP-FLOP.'*

The two inputs are: 'S', representing the Set input and 'R' the Reset input. The two outputs are: Q and $\bar{Q}$. $\bar{Q}$ is the inverse of Q. When Q is 'high' $\bar{Q}$ is 'low' and vice versa.

High Input Activated

Figure 2.12 shows the circuit symbol of a *FLIP-FLOP* which is activated when one of its inputs is taken 'high'.

Fig. 2.11 The low input activated R-S FLIP-FLOP circuit symbol.

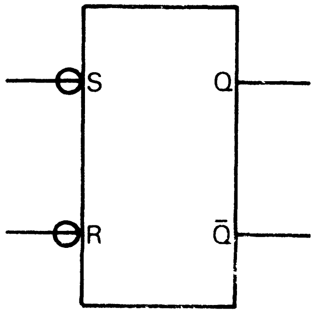
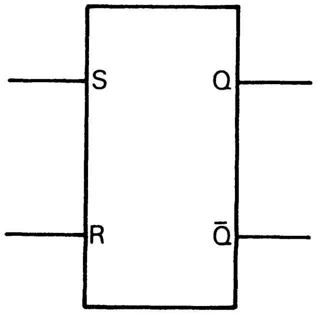


Fig. 2.12 The high input activated R-S FLIP-FLOP circuit symbol.



Low input activated FLIP-FLOP

Set condition

When the *FLIP-FLOP* is in the Set condition the Q output will be continually 'high', as shown in figure 2.13.

Reset condition

The *FLIP-FLOP* is in the Reset condition when the Q output is continually 'low', as shown in figure 2.14.

Activating the Flip-Flop

The inputs are normally resting 'high'.

To operate the Set input; it is taken 'low' as shown in figure 2.15. The Q output will go 'high', and remain so, even if the Set input returns 'high'.

Figure 2.16 shows a Reset command fed to the *FLIP-FLOP.* The Q output will go 'low', and stay 'low' until a Set pulse is received. Any further Reset commands will not affect the output.

Fig. 2.13 Set condition: Q continually 'high'.

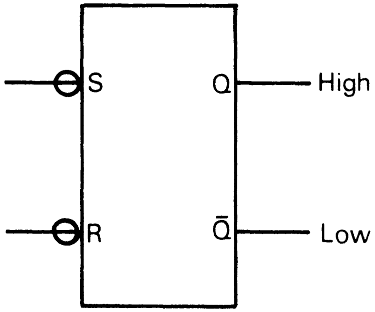


Fig. 2.14 Reset condition: Q continually 'low'.

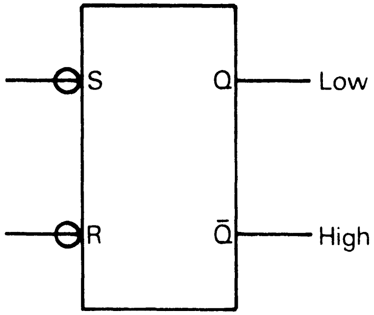


Fig. 2.15 Activating the Set input.

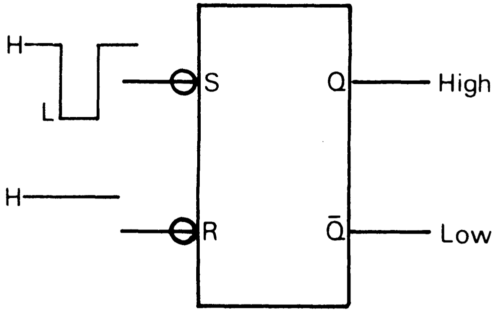
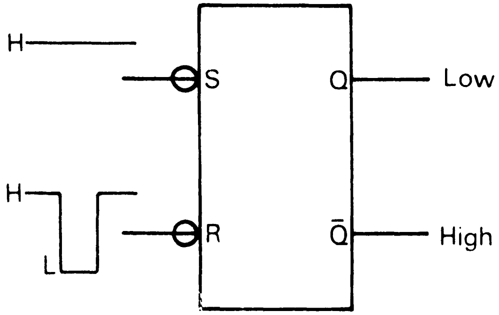


Fig. 2.16 Activating the Reset input.



High input activated FLIP-FLOP

Set condition

When the *FLIP-FLOP* is in the Set condition the Q output will remain 'high', as shown in figure 2.17.

Reset Condition

The *FLIP-FLOP* is in the Reset condition when the Q output is continually 'low', as shown in figure 2.18.

Activating the Flip-Flop

The inputs are normally resting 'low'.

To operate the Set input; it is taken 'high', as shown in figure 2.19. The Q output will go 'high', and remain 'high', even if the Set input returns 'low'.

Figure 2.20 shows a Reset command fed to the *FLIP-FLOP.* The Q output will go 'low', and stay 'low', until a Set pulse is received.

Fig. 2.17 Set condition: Q continually 'high'.

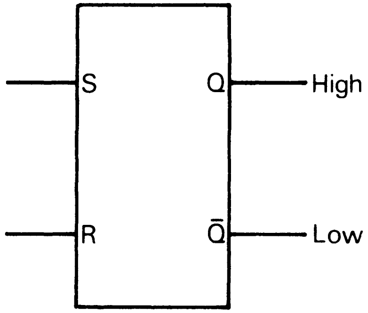


Fig. 2.18 Reset condition: Q continually 'low'.

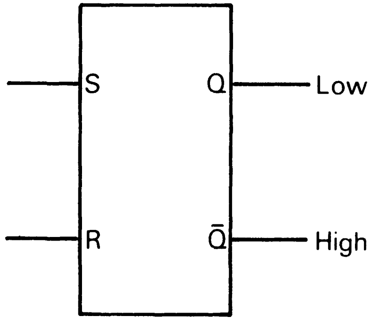


Fig. 2.19 Activating the Set input.

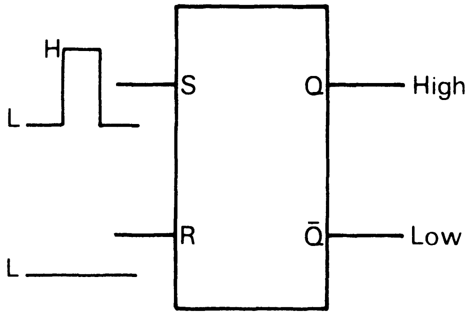
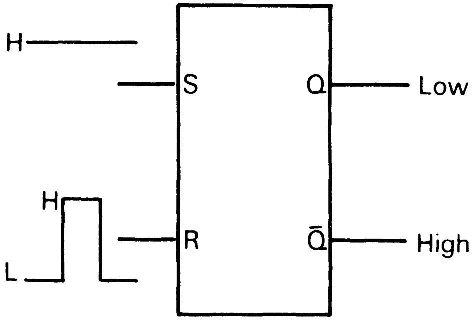


Fig. 2.20 Activating the Reset input.



Dynamic Operation

Figure 2.21 shows the 'High input activated *R-S FLIP-FLOP'* under a typical operating condition.

As can be seen from the diagram: the Q output changes, on the positive going edge of the input pulses. This type of diagram is known as a 'Timing diagram.'

Figure 2.22

The diagram figure 2.22, shows the operation of a 'Low input activated *R-S FLIP-FLOP'.* In this case: the Q output changes, on the negative going edge of the input pulses.

Fig. 2.21 Timing diagram; high input activated R-S FLIP-FLOP.

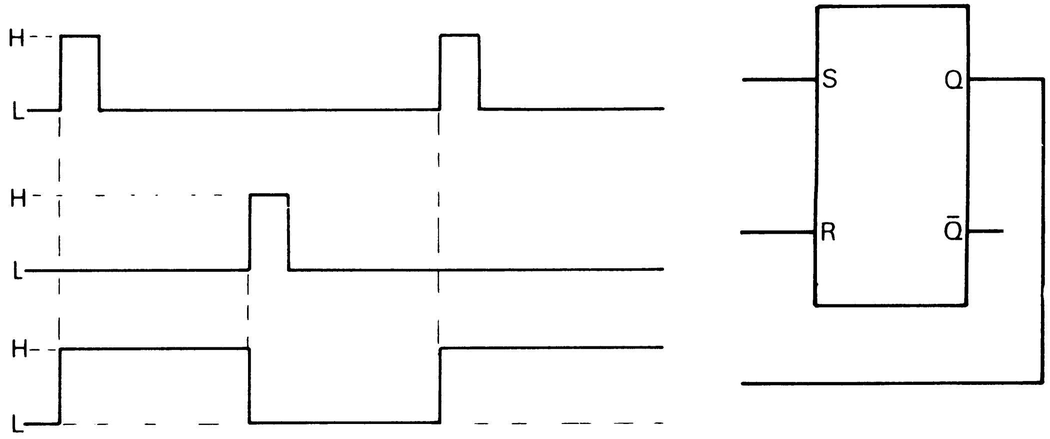
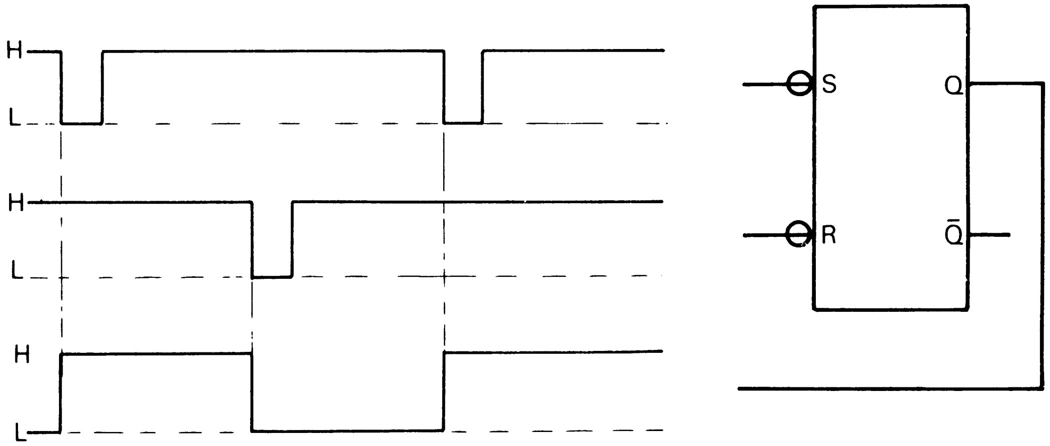


Fig. 2.22 Timing diagram; low input activated R-S FLIP-FLOP.



Nand Gate Flip-Flops

Figure 2.23 shows the symbol of a *FLIP-FLOP* which is activated by 'low' input pulses.

Figure 2.24 shows the internal configuration of this *FLIP-FLOP*.

INITIAL STATE (FIGURE 2.24)

Gate X has both inputs 'high' and so its output is 'low'. This 'low' is fed to one input of gate Y, making its output 'high', and so maintaining one of the two 'highs' on the input of gate X.

SET CONDITION (FIGURE 2.25)

The Set condition is where the Set input is taken 'low' and the Reset input is 'high'.

The Set input taken 'low' for a short time causes the output of gate X to go 'high'. Both inputs 'high' on gate Y will cause its output to go 'low'. This 'low' is fed to gate X, maintaining its output 'high', *EVEN WHEN THE 'LOW' ON THE SET INPUT IS NOW REMOVED.* Therefore with Set and Reset now both returned to 'high', the previous command (Set) is *'REMEMBERED'* (latched) with the Q output 'high' and the $\bar{Q}$ output 'low'.

RESET CONDITION (FIGURE 2.26)

The Reset condition is where the Set input is 'high' and the Reset input is taken 'low'.

The Reset input taken 'low' for a short time causes the output of gate Y to go 'high'. Both inputs 'high' on gate X will cause its output to go 'low'. This 'low' is fed to gate Y, maintaining its output 'high', *EVEN WHEN THE 'LOW' ON THE RESET INPUT IS NOW REMOVED.* Therefore with Set and Reset now both returned to 'high', the previous command (Reset) is *'REMEMBERED'* (latched) with Q output 'low' and $\bar{Q}$ output 'high'.

Fig. 2.23 The low input activated R-S FLIP-FLOP circuit symbol.

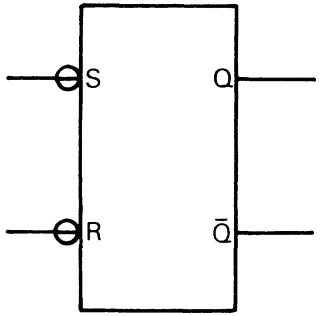


Fig. 2.24 Both Set and Reset inputs 'high'.

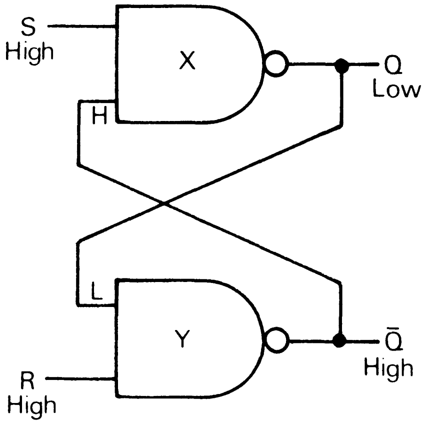


Fig. 2.25 Set input 'low', Reset input 'high'.

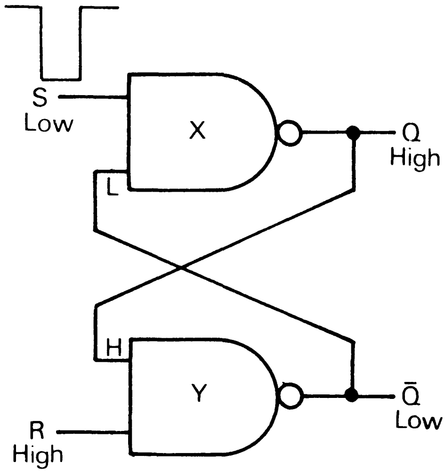
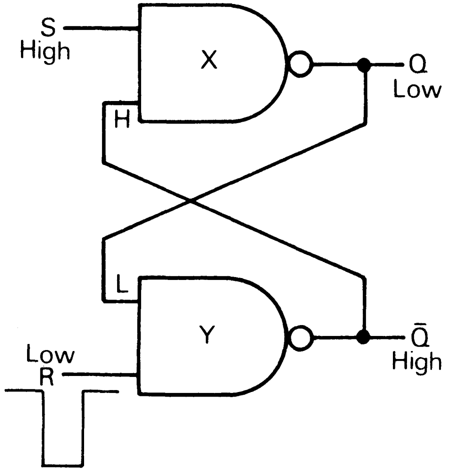


Fig. 2.26 Set input 'high', Reset input 'low'.



SET COMMAND FED TO A PREVIOUSLY SET FLIP-FLOP.

If a *FLIP-FLOP* is in the Set condition with Q 'high' and $\bar{Q}$ 'low' and a Set command is received then the *FLIP-FLOP* will stay as it is. Likewise if a *FLIP-FLOP* is in the Reset condition, reception of Reset commands will cause no change in the output.

APPLICATION.

The *'NAND FLIP-FLOP'* could be used to identify which of two lines has information on it. If line B has information on it, Q will be 'low', as shown in figure 2.27. If line A has information on it, Q will go 'high'.

SET AND RESET INPUTS BOTH TAKEN LOW.

In figure 2.28, if both R and S are taken 'low' at the same time, Q and $\bar{Q}$ will both go 'high'. This state is called *'UNDETERMINED'* and is avoided as, when both Set and Reset are returned 'high' together, one output will go 'high', and the other output will go 'low', but which output goes 'high' is random, hence the name undetermined.

Fig. 2.27 A practical application.

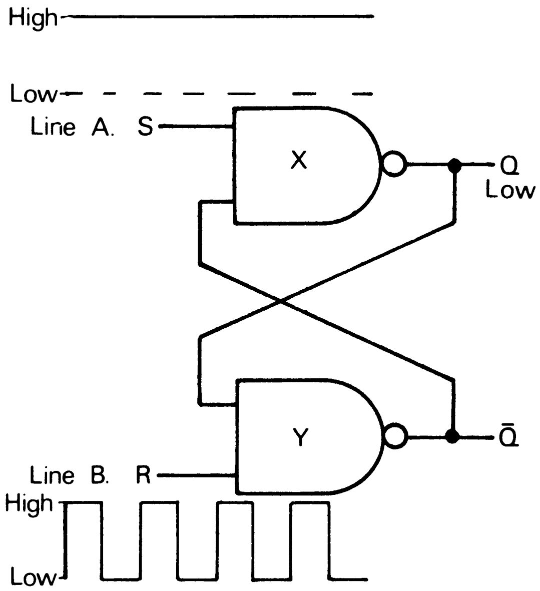


Fig. 2.28 Both Set and Reset inputs taken low. The 'undetermined' state.

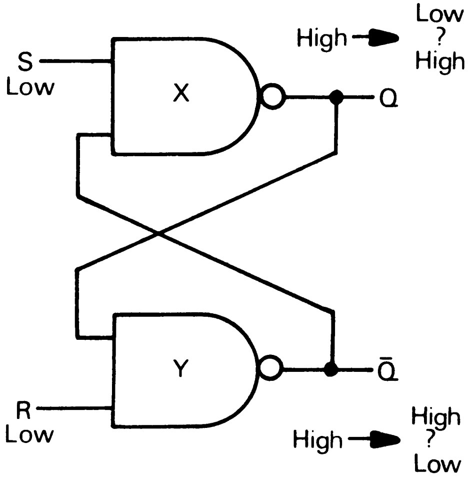


Figure 2.29

The diagram shows the circuit symbol of the 'High input activated *R-S FLIP-FLOP*.

Internal Configuration

Figure 2.30 shows the internal configuration of the 'High input activated *R-S FLIP-FLOP*'. It uses the **'Low activated *R-S FLIP-FLOP*' discussed on page 42.**

The inputs are inverted by g1 and g2, which means: a 'high' level input pulse will activate the *FLIP-FLOP*.

An alternative arrangement for a 'High activated *R-S FLIP-FLOP*' is given in the Appendix, page 61.

Practical Application

The 'High input activated *R-S FLIP-FLOP*' could be used in a channel change integrated circuit.

Refer to figure 2.31. The *FLIP-FLOP* is activated by taking S 'high' via a resistor chain and channel change switch.

Q going 'high' feeds the channel 1 tuning potentiometer.

When any other channel is required the Reset input is activated via g1.

When channel 1 is selected: the Q output of FF1 will be 'high'; the other five *FLIP-FLOPS* will have their Q outputs 'low'. If channel 2 is now selected: the 'high' voltage will be applied to the Set input of FF2, causing its Q output to go 'high'. At the same time the 'high' is fed, via g1, to the Reset input of FF1, and its Q output goes 'low'. FF2 now gives an output to the channel 2 tuning potentometer.

In practice; 'Low input activated *FLIP-FLOPS*' are used but the principle remains the same.

Fig. 2.29 The high input activated R-S FLIP-FLOP circuit symbol.

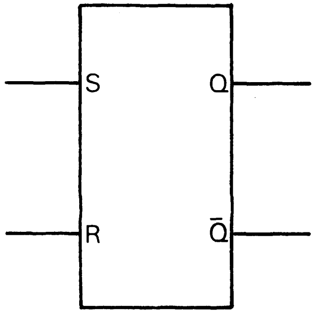


Fig. 2.30 Internal configuration of a high input activated R-S FLIP-FLOP.

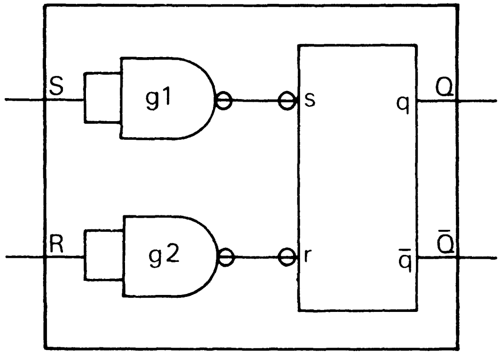
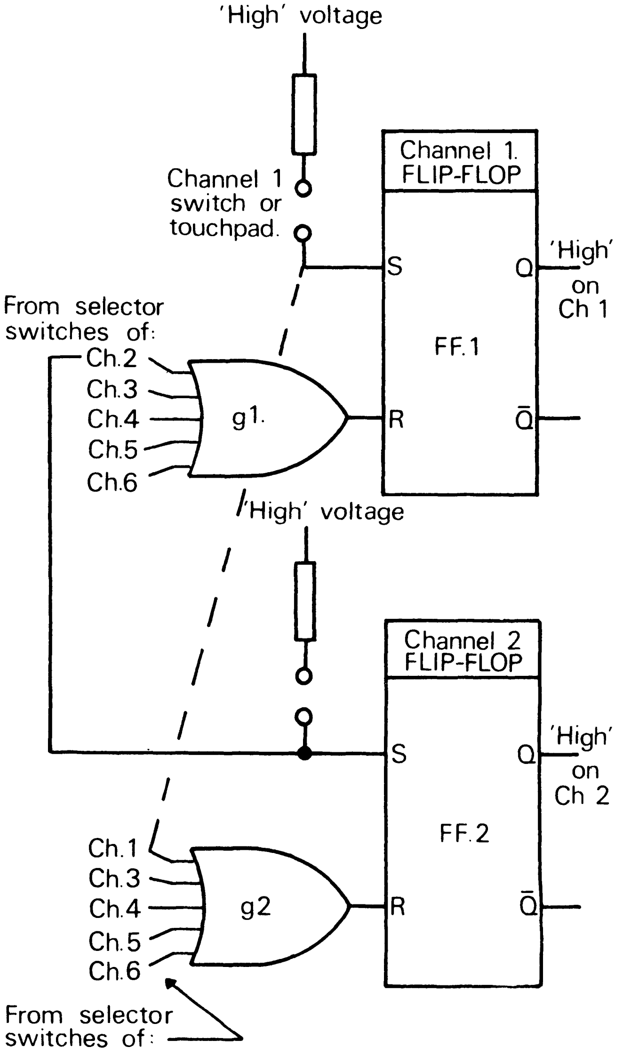


Fig. 2.31 A practical application.



Clocked R-S FLIP-FLOP

Frequently, it is desirable to set up the input conditions, but delay the actual Setting or Resetting action until a pulse is received from another source. This pulse is called the *'CLOCK'* pulse. Circuitry is added to the simple *FLIP-FLOP* for this facility.

CIRCUIT SYMBOL (FIGURE 2.32).

As before, the two inputs are Set and Reset. However any condition placed on S and R will have no effect on the Q output until the CK (Clock) input is taken 'high'.

TRUTH TABLE (FIGURE 2.33)

If the CK input is 'low', an input could be applied to the R and S inputs without affecting the Q and Q̄ outputs.

If the CK input is taken 'high' the truth table is similar to that of the *R-S FLIP-FLOP.* The Set and Reset inputs need a positive input to operate them.

CIRCUIT OPERATION.

In figure 2.34 the clock pulse is fed to the *FLIP-FLOP* via the extra circuitry g1 and g2. The gates g1 and g2 stop the data on S and R entering the *FLIP-FLOP* whilst the clock pulse is 'low'. With the clock pulse 'low' the outputs from g1 and g2 are 'high', with both inputs to the *FLIP-FLOP* 'high', it remembers the data entered previously. When the clock pulse goes 'high' g1 and g2 will allow the data on R and S to enter the *FLIP-FLOP.* If R is 'high' g2 output will be 'low'. 'Low' on R will cause the *FLIP-FLOP* to reset and Q will go 'low'.

Fig. 2.32 The clocked R-S FLIP-FLOP circuit symbol.

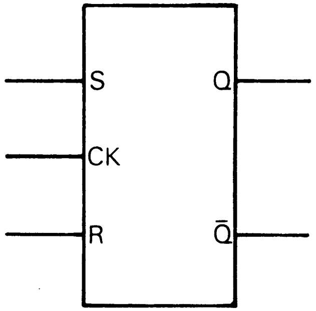
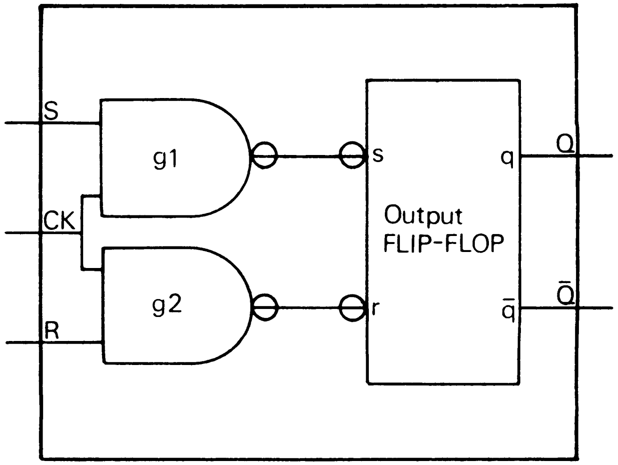


Fig. 2.33 The truth table.

Input			Output	
CK	R	S	Q	Q̄
H	L	L	No change	
H	L	H	H	L
H	H	L	L	H
H	H	H	H	H
H	But when both go Low.		H ? L	L ? H
L	X	X	No change	

X = Any input

Fig. 2.34 Circuit operation.



Clocked 'D' type FLIP-FLOP

The circuit symbol of a D type *FLIP-FLOP* is shown in figure 2.35.

The D type *FLIP-FLOP* is a variation of the clocked *R-S FLIP-FLOP.* It has one input, the D (Data) input. It also eliminates the undetermined state present in the *R-S* and the clocked *R-S FLIP-FLOP.*

Any data on the D input will have no effect on the Q outputs until the CK input is taken 'high'.

TRUTH TABLE (FIGURE 2.36)

If the CK input is 'low', any signal applied to the D input will have no effect on Q or Q̄.

If the CK input is taken 'high', the information that is on D will appear on Q, (and Q̄ will be the inverse of Q and D).

CIRCUIT OPERATION.

In figure 2.37, the D input goes directly to the Set input, and via the inverter gate (g1) to the Reset input. A 'high' on D is fed to the Set input directly and, via g1, becomes a 'low' on the Reset input. This will set the Q output 'high' on receipt of a clock pulse. A 'low' on D is fed directly to the Set input and, via gate gl, will make r 'high' resetting the Q output 'low' on *receipt of a clock pulse. The undetermined state (where set is 'high' and reset is 'high'), cannot occur.*

D TYPE FLIP-FLOP APPLICATIONS.

The D type *FLIP-FLOP* shown in figure 2.38 transfers data from the D terminal to Q, when its CK input is 'high'. The data will stay on the Q output and be remembered when the clock is taken 'low' and stays 'low'.

This can be used to 'freeze' the output until the next piece of data that is required is entered by raising the CK input again.

Fig. 2.35 The 'D' type FLIP-FLOP circuit symbol.

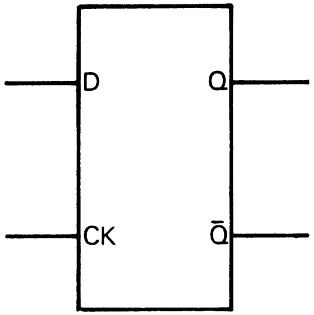


Fig. 2.36 The truth table.

Input		Output	
CK	D	Q	Q̄
H	L	L	H
H	H	H	L
L	X	No change	

X = Any input

Fig. 2.37 Circuit operation.

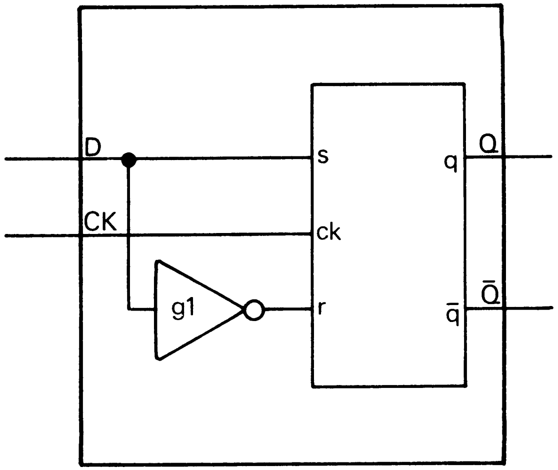
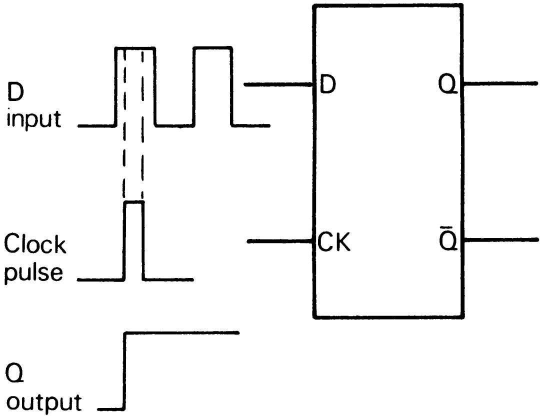


Fig. 2.38 A practical application.



Edge triggered FLIP-FLOPS

The D type *FLIP-FLOP* described on the previous page reacts to its input when the clock pulse is 'high'. If the input data altered during the clock pulse the output data would alter. The data would therefore feed straight through the *FLIP-FLOP* (often called ripple through), and affect circuits that it should not. To prevent this; when data is to be entered, the input terminal will have to stay at the required level for the clock pulse duration (as shown in figure 2.39). This will limit the rate at which the data changes.

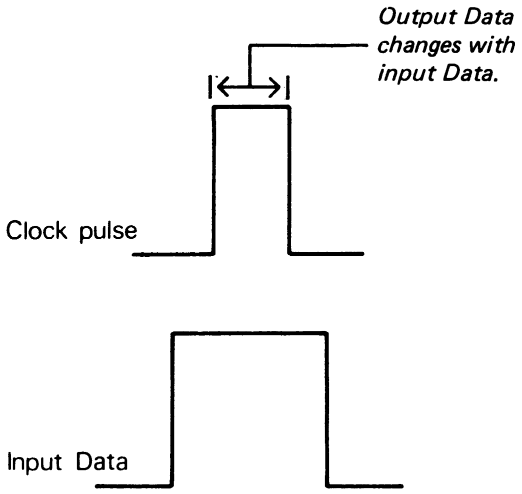
EDGE TRIGGERING (FIGURE 2.40).

To overcome the problem outlined above, edge triggered *FLIP-FLOPS* are available. They transfer the input terminal data on the edge of the clock pulse, and then once data has entered, will disconnect the input terminal so it has no effect on the output.

CIRCUIT SYMBOL.

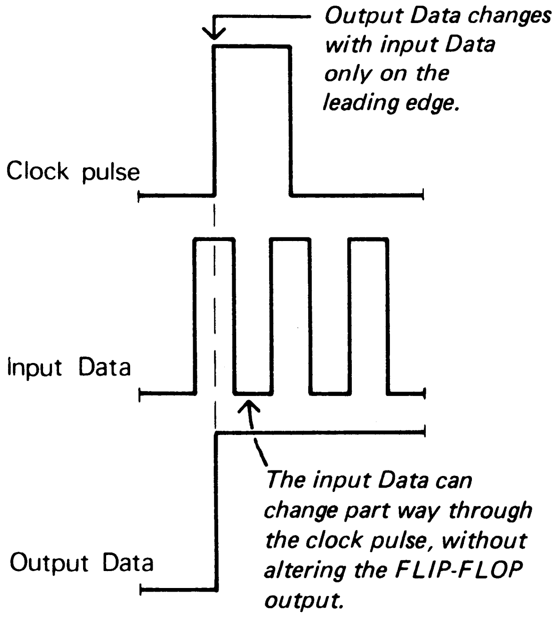
The edge triggered D type circuit symbol is shown in figure 2.41. The arrow on the CK input indicates that it is positive edge triggered. That is; data enters on the positive going edge of the clock pulse.

Fig. 2.39 Comparing the clock pulse to input Data.



Output Data changes with input Data.

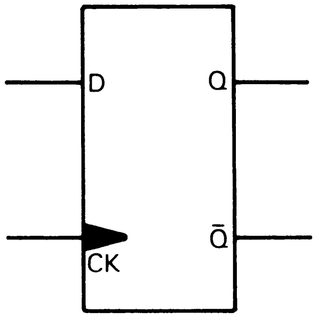
Fig. 2.40 Edge triggering.



Output Data changes with input Data only on the leading edge.

The input Data can change part way through the clock pulse, without altering the FLIP-FLOP output.

Fig. 2.41 The edge triggered 'D' type circuit symbol.



EDGE TRIGGERED D TYPE FLIP-FLOPS

Reference figure 2.42. This is a differentiating circuit. If the time constant (CR seconds) is much smaller than the clock pulse width, then the output waveform is a series of narrow pulses.

The positive narrow pulses enable the AND gates (figure 2.43) for a short period, the negative pulses do nothing.

During the positive pulses the data at the D input terminal is fed via g1 and g2 to activate the flip-flop.

When D = 1: g 1 is enabled, S = 1 and R = 0 therefore Q = 0.

This type of triggering is called 'edge triggering' because the flip-flop only responds when the inputs to the AND gates are high, i.e. for the short period during the leading edge of the clock pulse.

The flip-flop could also be designed to operate on the negative pulse, i.e. the flip-flop would respond to the trailing edge of the clock pulse.

Fig. 2.42 The principle of edge triggering.

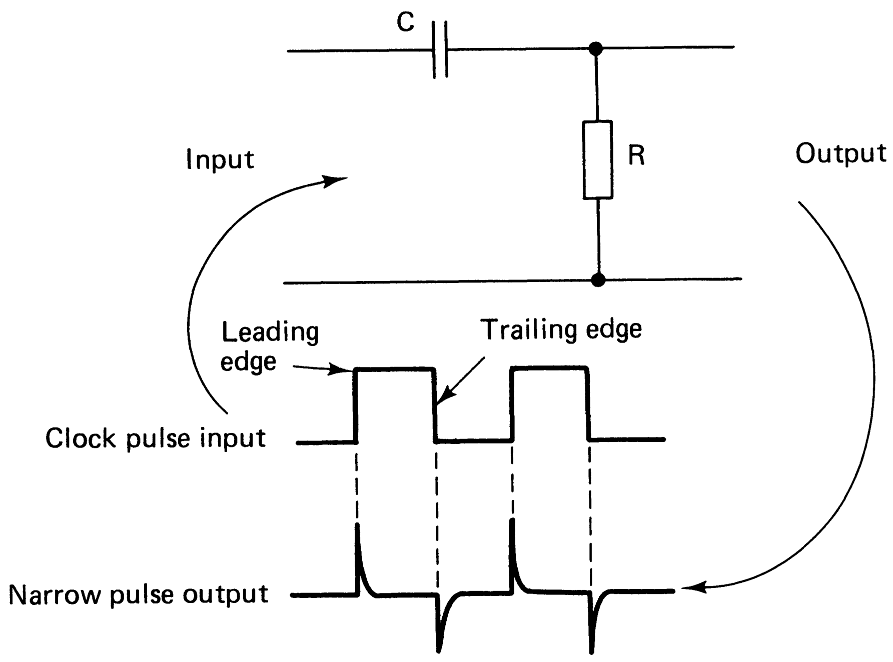
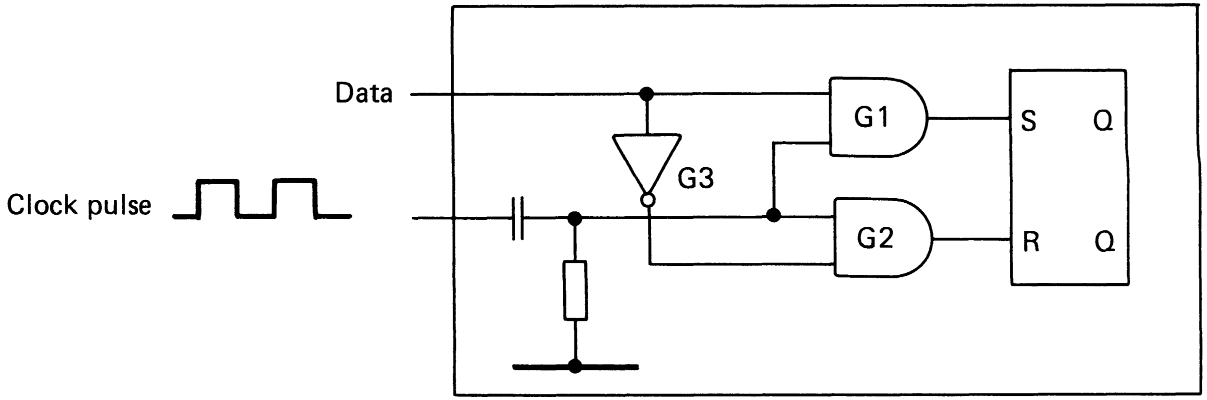


Fig. 2.43 Circuit operation.



The toggle FLIP-FLOP

In a large number of applications, the outputs are required to change state with each clock pulse, as in the U704 circuit discussed on page 37. The change on each clock pulse may need to be delayed until required. This is the purpose of the T terminal on the toggle *FLIP-FLOP.*

CIRCUIT SYMBOL.

The circuit symbol of the toggle *FLIP-FLOP* is shown in figure 2.44. In this case there is a T input that controls the output. When the T input is 'low' then Q and $\bar{Q}$ will remain unchanged, even if the *FLIP-FLOP* is clocked. When T is taken 'high', each time a clock pulse arrives the outputs will change state on the leading edge of the clock pulse.

TRUTH TABLE (FIGURE 2.45).

The truth table shows that if the T input is 'low' the Q outputs will remain unchanged. With the T input 'high' the Q outputs will not change until the clock goes 'high'.

On the leading edge of the clock pulse, the outputs change state (Toggle).

Fig. 2.44 The toggle FLIP-FLOP circuit symbol.

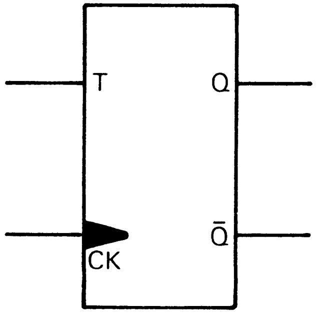


Fig. 2.45 The truth table.

Input		Output	
CK	T	Q	$\bar{Q}$
L	L	No change	
H	L	No change	
L	H	No change	
H	H	Toggles	

CIRCUIT OPERATION.

With the D type *FLIP-FLOP*, the Q output when clocked, will go to the state presented at the Data terminal. If the Data terminal is connected to $\bar{Q}$ (as shown in figure 2.46a), on each clock pulse leading edge, Q will go to the state $\bar{Q}$ was before the clock pulse. That is, the outputs will change state (toggle).

If the Data terminal is connected to Q (Figure 2.46b) then on each clock pulse leading edge, Q will go to Q. That is, it will remain the same.

Figure 2.47a shows how an *AND* gate may be used to prevent the flow of Data.

Whilst input B is held 'low', the output will also be 'low', regardless of the level at the Data terminal A. This process is known as *'inhibiting'* or the data is said to be inhibited.

Figure 2.47b shows that the *AND* gate will pass data when input B is 'high', then, whatever is on the other input will appear at the output. When data is allowed to pass through a circuit that circuit is said to be *'enabled'.*

If input B is 'high' the data is enabled.

FIGURE 2.48

The $\bar{Q}$ output is coupled to the D input via gate X.

The Q output is coupled to the D input via gate Y.

With the toggle input 'high', gate X is enabled. The $\bar{Q}$ output is coupled to the D input via gates X and Z; this causes the FLIP-FLOP to toggle on each clock pulse leading edge.

With the toggle input 'low', gate X is inhibited, and will give a 'low' output to gate Z. The 'low' on T is inverted to 'high' by gate W, which enables gate Y. The Q output is connected to the D input via gates Y and Z. On each clock pulse the Q output will remain unchanged, i.e., no toggling takes place.

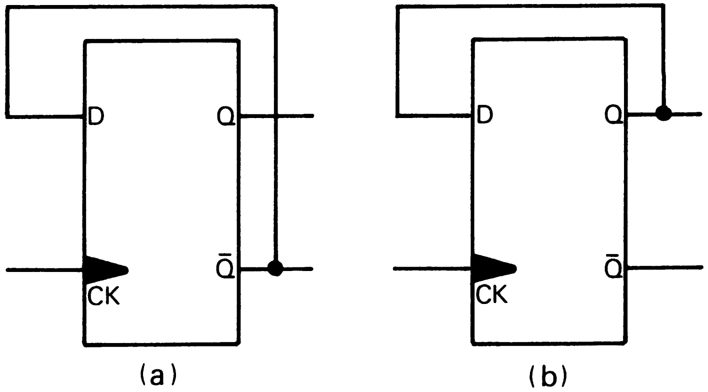
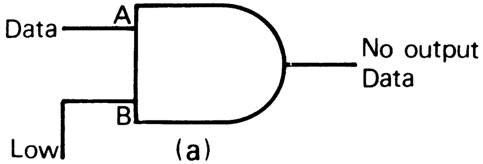
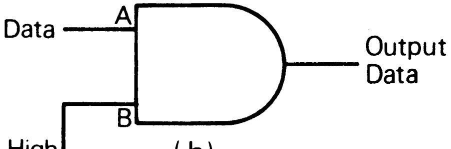


Fig. 2.46 The 'D' type FLIP-FLOP connected in (a) as a toggle and in (b) to prevent toggle action.



(a)
INHIBITING



(b)
ENABLING

Fig. 2.47 The AND gate used to control the flow of Data

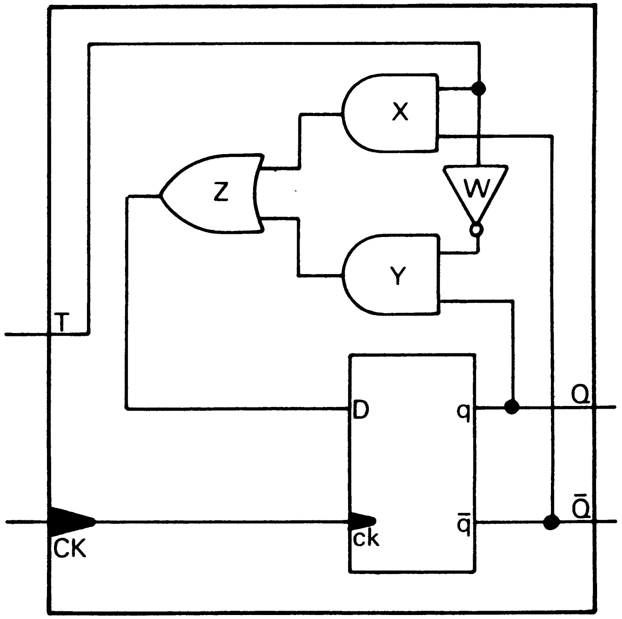


Fig. 2.48 Circuit operation.

TOGGLE FLIP-FLOP APPLICATIONS.

The toggle *FLIP-FLOP* can be used to divide a signal by two. The signal to be divided is fed to the CK input. If the T input is held 'high' the divided signal can be taken from Q or $\bar{Q}$.

Each time the input goes 'high' the Q output will change (Figure 2.49). Therefore two cycles of input signal will produce one cycle at each of the two outputs. The input signal frequency has been halved or divided by two.

The toggle *FLIP-FLOP* can be used to produce a square waveform from a waveform of uneven mark-space ratio.

As may be seen in figure 2.50, the output change takes place on the positive going edge of the clock input. As the positive going edges are equally spaced, each output half cycle will have the same duration, and so a square waveform results.

When the toggle terminal is taken 'low' no change in output occurs.

Fig. 2.49 Dividing a signal by the factor of 2.

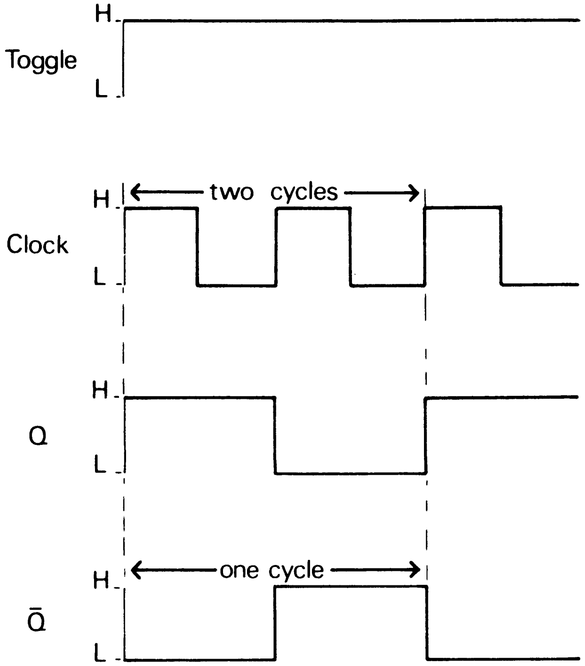
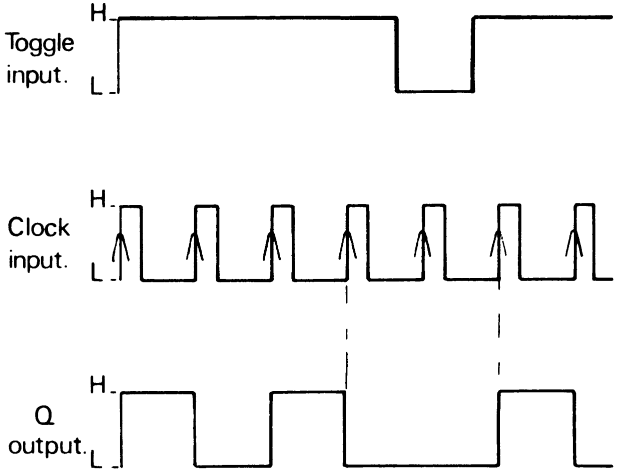


Fig. 2.50 Producing a squarewave from a signal with an uneven mark-space ratio.



Negative edge triggering

On page 47 positive edge triggering was described, and it was seen that once the *FLIP-FLOP* 'took up data', the data input could change to a new state. Some *FLIP-FLOPS* take in data on the negative going edge of the clock pulse. These are called 'negative edge triggered' devices.

In figure 2.51 'positive edge triggered' devices would take in the 'high' data. 'Negative edge triggered' devices would take in the 'low' data.

CIRCUIT SYMBOL (FIGURE 2.52).

A negative edge triggered D type *FLIP-FLOP* circuit symbol is shown. The arrow on the CK input indicates that it is edge triggered. The circle indicates that it is negative edge triggered. The truth table will be similar to the clocked D type *FLIP-FLOP* on page 46 but the output would change at the negative going edge.

TIMING DIAGRAM (FIGURE 2.53)

When the data input signal changes, this appears on the Q output at the negative going edge of the clock pulse.

Fig. 2.51 Edge triggering

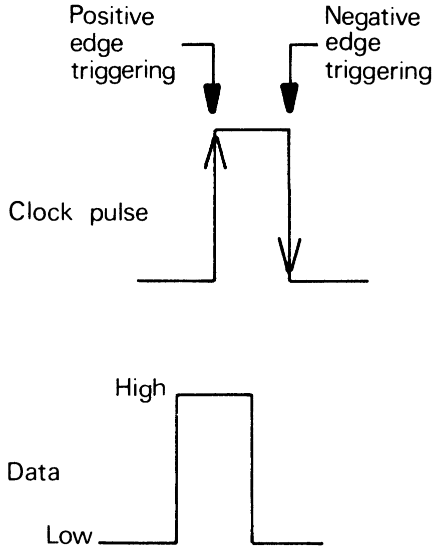


Fig. 2.52 The negative edge triggered 'D' type FLIP-FLOP circuit symbol.

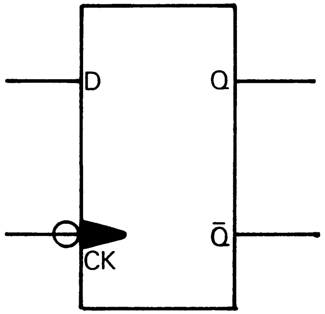
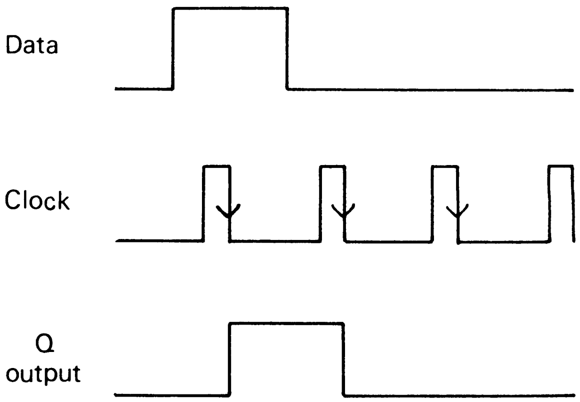


Fig. 2.53 Timing diagram.



CIRCUIT OPERATION.

Refer to figure 2.54. Two *FLIP-FLOPS* are used. They are called *MASTER* and Slave *FLIP-FLOPS.* The package is known, as a *MASTER SLAVE D TYPE FLIP-FLOP.*

Both 'Master' and 'Slave' *FLIP-FLOPS* require a 'high' on their ck inputs to transfer the data from their d input, to their q output.

The clock pulse on the CK input is fed directly to the 'Master' and is inverted to feed the 'Slave'. When the CK input is 'low' (prior to a clock pulse); the 'Master' is inhibited; data on its d input cannot enter; the 'Slave' receives a 'high' and is enabled; whatever is on the 'Master q' output (entered on the previous clock pulse) is passed to the 'Slave' q output.

When the CK input is 'high' (during the clock pulse); the 'Master' is enabled; data enters and appears on its 'q' output; the 'Slave' clock receives a 'low' and is inhibited; data that is on the 'Master q' output cannot pass through to the 'Slave' q output.

As the CK input goes 'low' the 'Master' is inhibited. The data that was presented to the D input just prior to the clock going 'low' is remembered by the 'Master', and is on the 'Master q' output. The 'Slave' clock goes 'high' slightly after the 'Master' is inhibited. It is now enabled, and transfers the data on the 'Master q' to the 'Slave q' output. If the data on the main Q output is required to change state, it would do so at the negative going edge of the clock pulse.

SUMMARY

The D input signal is transferred to the 'q' output of the 'Master' during the positive edge of the clock pulse, and then transferred to the q output of the 'Slave' during the negative going edge of the clock pulse.

CLOCK TIMING DIAGRAM.

The clock pulse is often used to indicate the precise timing of the preceding operation. This is explained by the diagram in figure 2.55.

Fig. 2.54 Circuit operation.

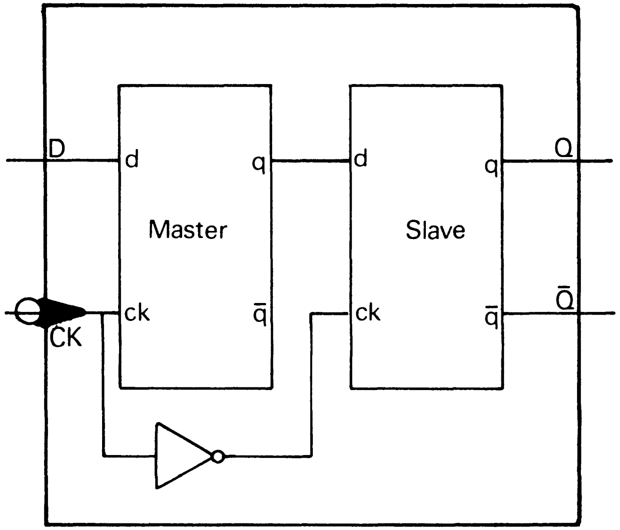
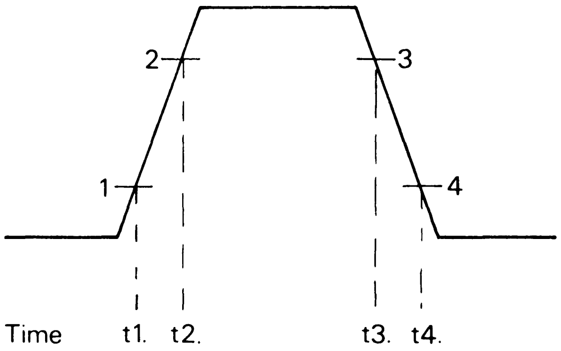


Fig. 2.55 Clock timing diagram.

1. *Isolate 'Slave' from 'Master', (to isolate the output from the 'Master').*
2. *Connect 'Master' to the D input: (data can now enter the 'Master').*
3. *Disconnect 'Master' from the D input: (data no longer enters 'Master').*
4. *Connect 'Master' to 'Slave': (to allow the 'Slave' to take up the data entered).*



The J-K master slave FLIP-FLOP

The J-K 'Master Slave' *FLIP-FLOP* is a very versatile device. It could be used (by external connections), to replace most of the *FLIP-FLOPS* described previously.

CIRCUIT SYMBOL (FIGURE 2.56).

The D type 'Master Slave' *FLIP-FLOP* shown on page 53 has only one input terminal. The *J-K FLIP-FLOP* has two input terminals which will make it more versatile. As the circuit symbol shows, it is negative edge triggered.

TRUTH TABLE.

The truth table of the J-K 'Master Slave' *FLIP-FLOP* is shown in figure 2.57.

The J input terminal if taken 'high' with K 'low', sets the Q output 'high' on the negative going edge of the clock pulse. The K input terminal if taken 'high' with J 'low', resets the Q output 'low' on the negative going edge of the clock pulse. If both J and K are 'low' the *FLIP-FLOP* is neither Set nor Reset and remains unchanged.

If both J and K are 'high', the *FLIP-FLOP* toggles on the negative going edge of each clock pulse.

Fig. 2.56 The J-K FLIP-FLOP circuit symbol.

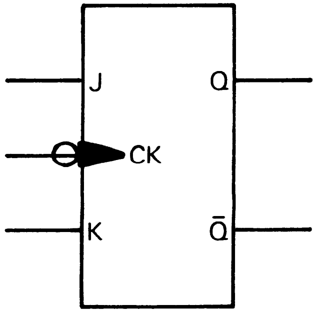


Fig. 2.57 The truth table.

Input		Output after clock	
J	K	Q	$\bar{Q}$
L	L	No change	
L	H	L	H
H	L	H	L
H	H	$\bar{Q}$	Q

CIRCUIT OPERATION.

The J-K 'Master Slave' *FLIP-FLOP* is shown in figure 2.58. It consists of two *FLIP-FLOPS* and, as was discussed with the D type 'Master Slave' *FLIP-FLOP,* the information is transferred to the output terminals on the negative going edge of the clock pulse.

To achieve the toggle action there are feedback paths from Q and Q̄ to the input, as was seen with the toggle *FLIP-FLOP* on page 50.

If J and K are 'high', gates 1 and 2 are enabled; the feedback path can operate, and toggling occurs.

TOGGLING (J AND K HIGH).

Refer to figure 2.59. Assume Q is 'low' and Q̄ is 'high'. The 'high' from Q̄ causes g1 output to be 'high', the 'low' from Q causes g2 output to be 'low'. The Master q output goes 'high' (Set) on the clock pulse positive going edge and the Slave q output goes 'high' on the clock pulse negative going edge. The Q output has therefore changed state on the negative going edge of the clock pulse.

Now try the exercise on the following page.

Fig. 2.58 The J-K master slave FLIP-FLOP.

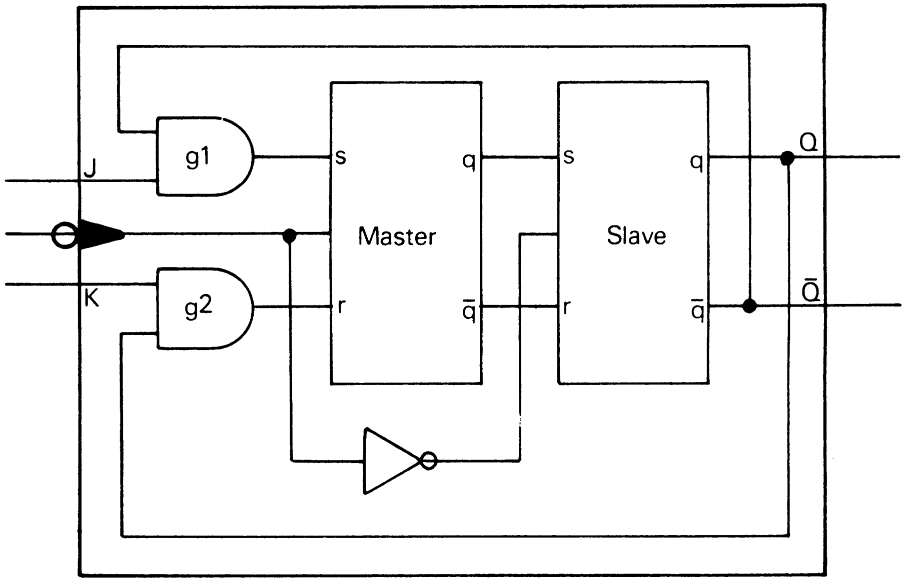
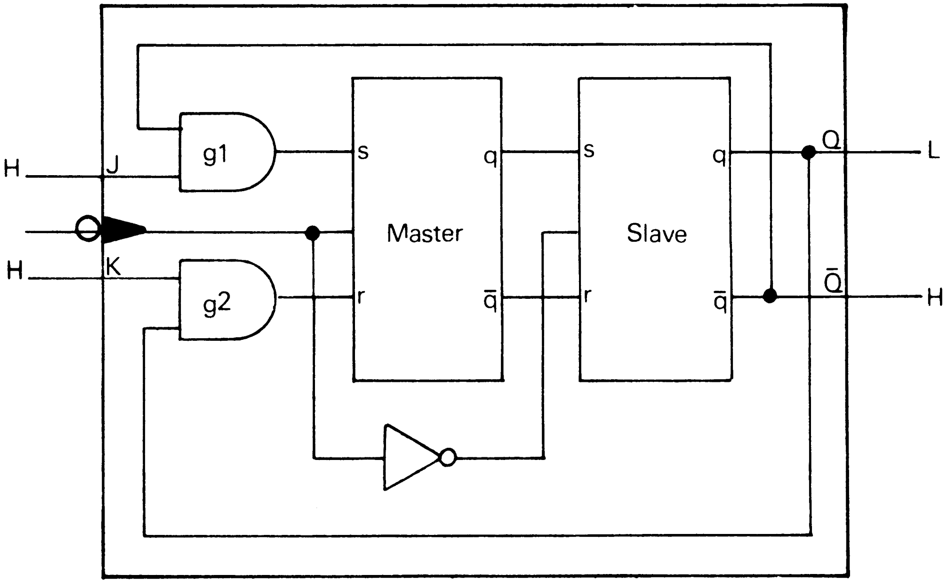


Fig. 2.59 Toggling.



J-K FLIP-FLOP EXERCISE.

The truth table in figure 2.61 refers to the *J-K FLIP-FLOP* in figure 2.60. The purpose of this exercise is to work with the permutations of J-K and Q before the clock pulse, to determine what the Q output will be after the clock pulse. The answer is given on page 59.

Fig. 2.60 The J-K FLIP-FLOP.

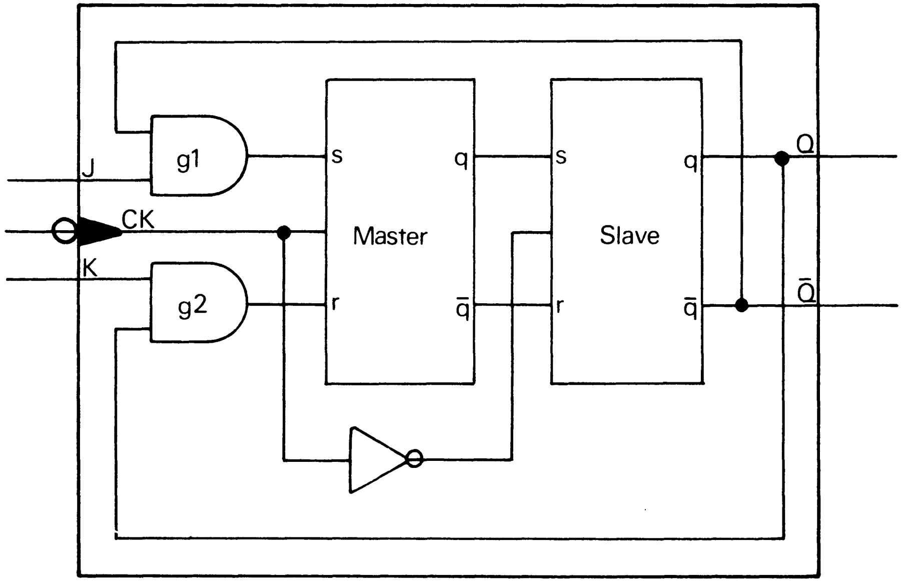


Fig. 2.61 Exercise 2.1. The solution is on page 59.

	Input		Q before Clock	Q after Clock
	J	K		
A	L	L	L	
B	L	L	H	
C	L	H	L	
D	L	H	H	
E	H	L	L	
F	H	L	H	
G	H	H	L	
H	H	H	H	

J-K master slave with set-reset facility

With the *J-K FLIP-FLOP* shown on the previous page, control of the outputs by J and K will occur on the negative going edge of the clock pulse. When inputs have no effect until the required clock pulse condition occurs, they are called 'Synchronous' inputs. Their effect is in synchronism with the clock pulse.

SET AND RESET INPUTS.

The J-K *FLIP-FLOP* shown in figure 2.62 has Set and Reset inputs. If one of the inputs is taken 'low' they will immediately set Q 'high' or, reset Q 'low' irrespective of J-K and CK inputs. These inputs are called 'asynchronous' inputs as they do not work in synchronism with the clock pulse.

SET/RESET TRUTH TABLE. FIGURE 2.63.

With Set 'low' and Reset 'high' the Q output is set 'high'. With Set 'high' and Reset 'low' the Q output is reset 'low'.

CIRCUIT OPERATION. FIGURE 2.64.

The Set and Reset inputs have direct access to the 'Slave' output *FLIP-FLOP* and therefore over-ride the other inputs.

Fig. 2.62 The J-K FLIP-FLOP with Set and Reset inputs circuit symbol.

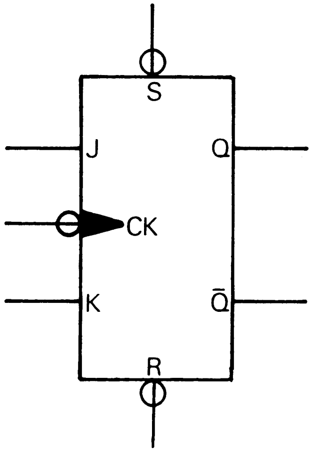
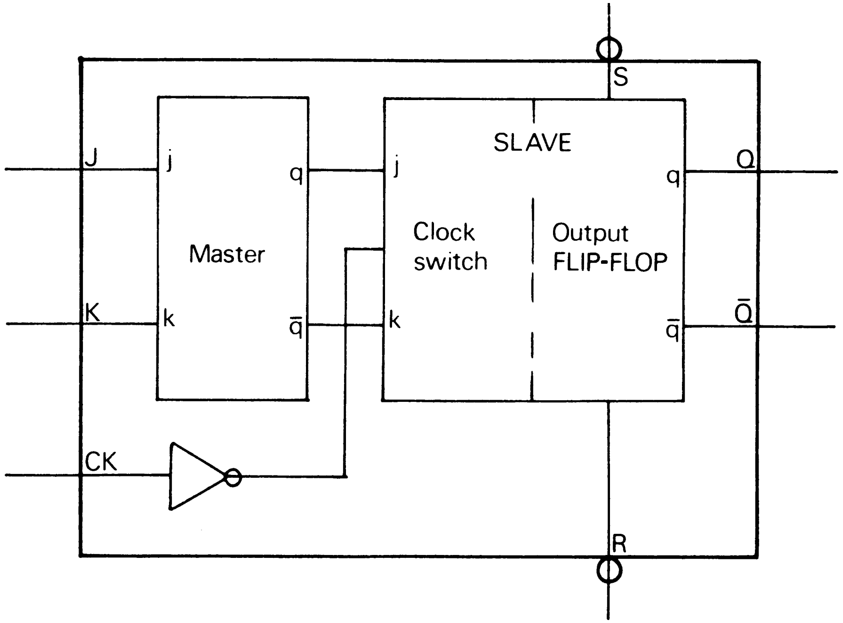


Fig. 2.63 Set Reset truth table.

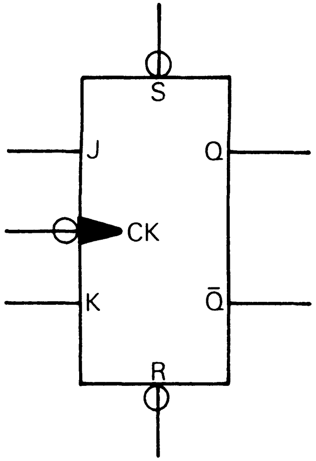
Input		Q
S	R	
L	H	H
H	L	L

Fig. 2.64 The J-K FLIP-FLOP with Set and Reset inputs.



EXERCISE 2.2

Figure 2.65 is an exercise. Draw in the Q output. The solution is on page 60.



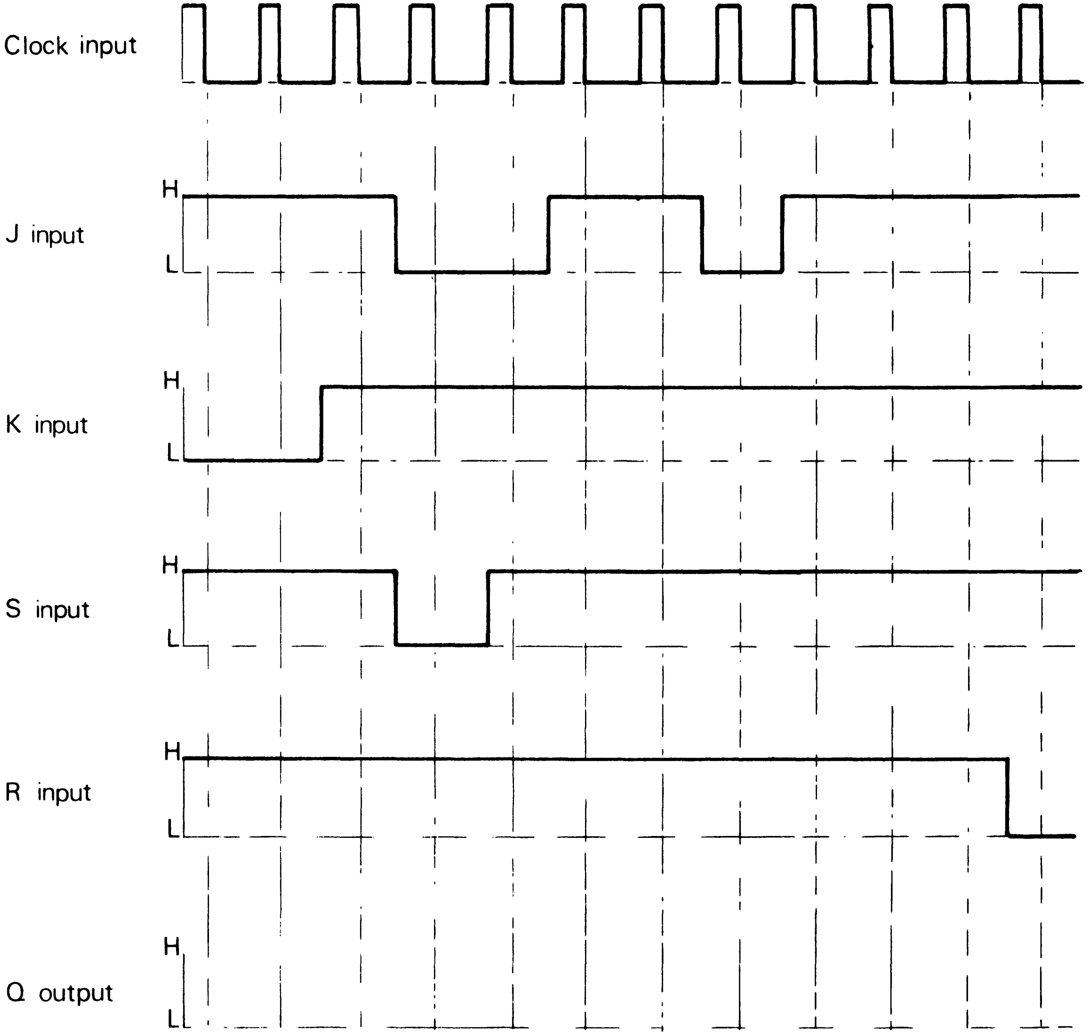


Fig. 2.65 Exercise 2.2. The solution is on page 60.

J-K FLIP-FLOP EXERCISE ANSWERS.

A). With J and K 'low' gates g1 and g2 both have at least one 'low' input and so their output is 'low'. The 'Master' *FLIP-FLOP* is therefore neither Set nor Reset and so the Q output remains unchanged.

B). For the same reason as A, the Q output remains unchanged.

C). Gate (g1) has a 'low' from J and a 'high' from $\bar{Q}$. Its output is 'low'. Gate (g2) has a 'high' from K and a 'low' from Q. Its output is 'low'. The *FLIP-FLOP,* therefore, is neither Set nor Reset and so stays in the Reset State with Q 'low'.

D). Gate (g1) has a 'low' from J and a 'low' from $\bar{Q}$. Its output is 'low'. Gate (g2) has a 'high' from K and a 'high' from Q. Its output is 'high'. Gate (g2) output going 'high' will Reset the *FLIP-FLOP* making Q go 'low' on the trailing edge of the clock pulse.

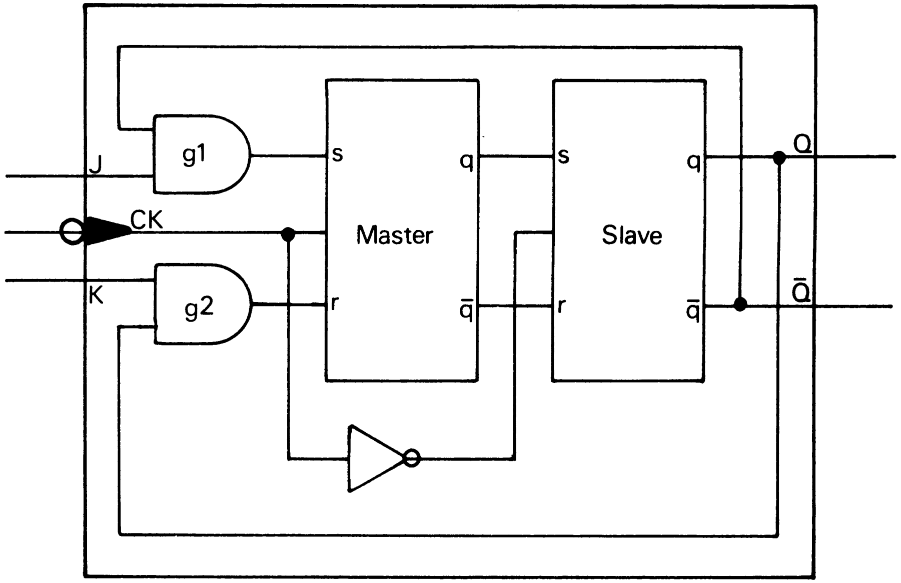
E). Gate (g1) has a 'high' output. Gate (g2) has a 'low' output. This will set the Q output 'high'.

F). Gates (g1) and (g2) both have a 'low' output and so the *FLIP-FLOP* stays in the Set state.

G). Gate (g1) has a 'high' output. Gate (g2) has a 'low' output. This sets the Q output 'high' on the trailing edge of the clock pulse.

H). Gate (g1) has a 'low' output. Gate (g2) has a 'high' output. This resets the Q output 'low' on the trailing edge of the clock pulse.

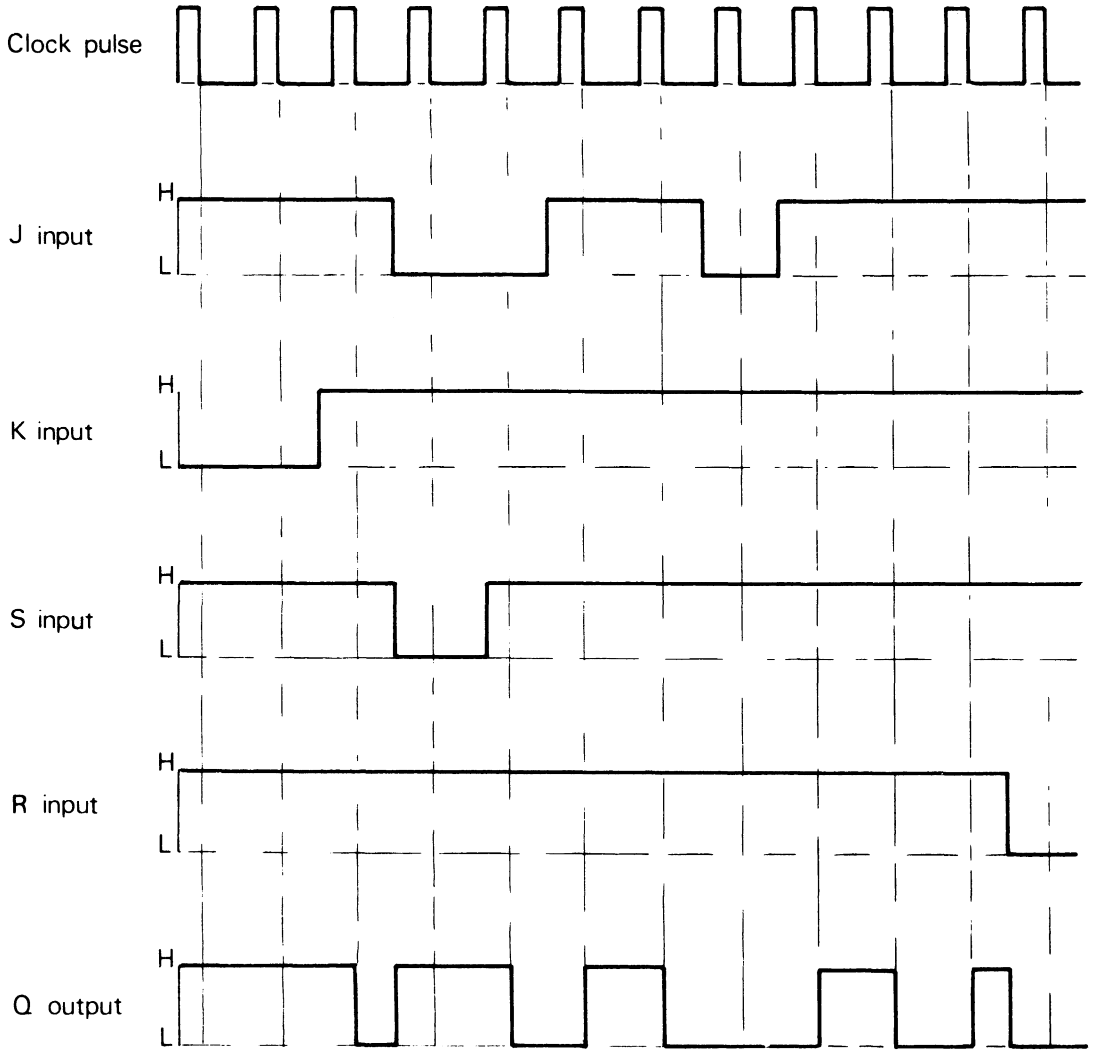
Fig. 2.66 Solution to exercise 2.1 (Fig. 2.61).



	Input		Q before Clock	Q after Clock
	J	K		
A	L	L	L	L
B	L	L	H	H
C	L	H	L	L
D	L	H	H	L
E	H	L	L	H
F	H	L	H	H
G*	H	H	L	H
H*	H	H	H	L

* J and K are high, causing a toggling action.

Fig. 2.67 Solution to exercise 2.2.



Appendix to Part 2

The *R-S FLIP-FLOP* may be constructed from cross-coupled *NOR* gates as shown in figure 2.68.

A HIGH INPUT ACTIVATED FLIP-FLOP.

The circuit symbol of the *'HIGH INPUT ACTIVATED FLIP-FLOP'* is shown in figure 2.69.

The two inputs are, 'S' standing for 'Set' input and, 'R' standing for 'Reset' input. The two outputs are: Q and $\bar{Q}$. $\bar{Q}$ is the inverse of Q. If Q is 'high' $\bar{Q}$ will be 'low'.

As there are no state indicator circles on the input, the input required to operate the *FLIP-FLOP* is a high going pulse.

Fig. 2.68 The NOR R-S FLIP-FLOP.

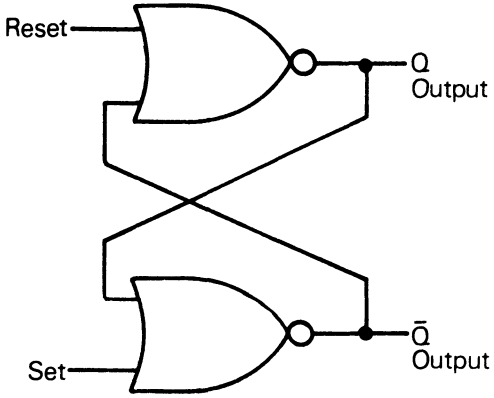
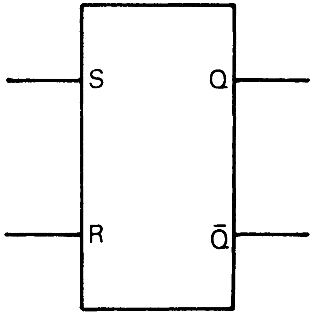


Fig. 2.69 The high input activated FLIP-FLOP circuit symbol.



NOR R-S FLIP-FLOP.

Figure 2.70 revises the input and output states for a *NOR* gate.

INITIAL STATE (FIGURE 2.71).

Gate Y has both inputs 'low' so its output is 'high'. This 'high' is fed to one input of gate X making its output 'low', therefore maintaining one of the two 'lows' on the input to gate Y.

SET CONDITION (FIGURE 2.72).

The 'set condition' is where the Set input is taken 'high' and the Reset input is 'low'.

The Set input taken 'high' for a short time causes the output of gate Y to go 'low'. Both inputs 'low' on gate X will cause its output to go 'high'. This 'high', is fed to gate Y, maintaining a 'low' output on gate Y, *EVEN IF THE 'HIGH' ON THE SET INPUT IS NOW REMOVED*. Therefore; with the Set and Reset now both returned to 'low', the previous command (Set) is *'REMEMBERED'* (latched), with the Q output 'high' and the $\bar{Q}$ output 'low'.

RESET CONDITION (FIGURE 2.73).

The Reset condition is where the Set input is 'low' and the Reset input is taken 'high'.

The Reset input taken 'high' for a short time causes the output of gate X to go 'low'.

Both inputs 'low' on gate Y will cause its output to go 'high'. This 'high' is fed to gate X maintaining its output 'low', *EVEN IF THE 'HIGH' ON THE RESET INPUT IS NOW REMOVED*. Therefore; with Set and Reset now both returned to 'low', the previous command (Reset) is *'REMEMBERED'* (latched), with the $\bar{Q}$ output 'high' and the Q output 'low'.

Fig. 2.70 Summary of the NOR gate.

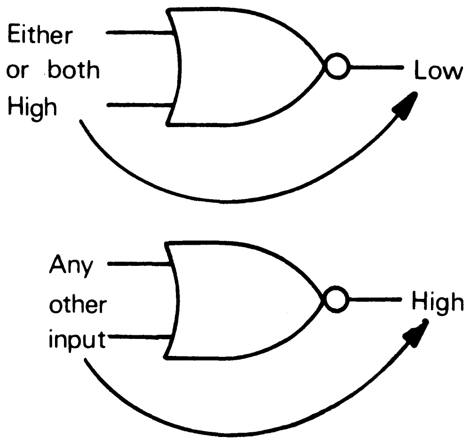


Fig. 2.71 NOR R-S FLIP-FLOP with Set input 'low' and Reset input 'low'.

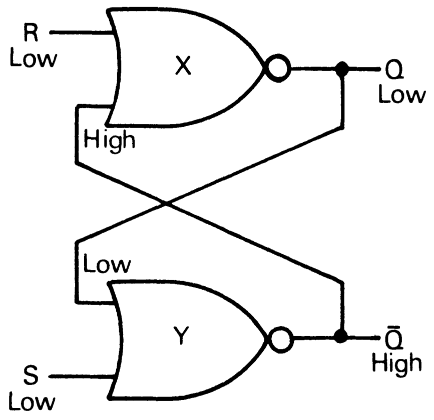


Fig. 2.72 Set input 'high' and Reset input 'low'.

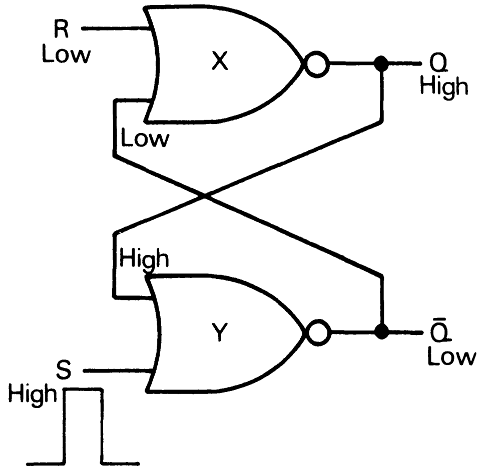
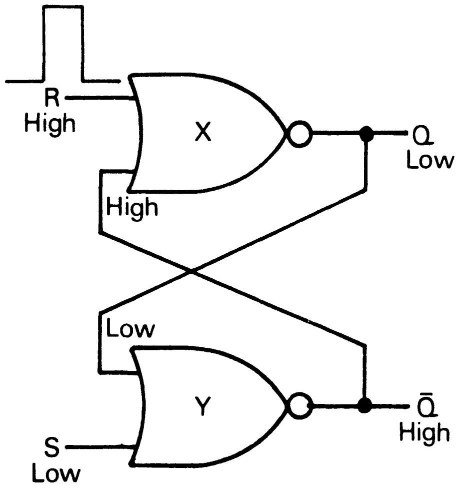


Fig. 2.73 Set input 'low' and Reset input 'high'.



SET COMMAND FED TO A PREVIOUSLY SET FLIP-FLOP.

If a *FLIP-FLOP* is in the Set condition with Q 'high' and $\bar{Q}$ 'low' and a Set command is received, then the *FLIP-FLOP* will stay as it was. Likewise, if a *FLIP-FLOP* is in the Reset condition, the reception of a Reset command will cause no change in the output.

APPLICATION.

The *FLIP-FLOP* mentioned could be used to identify which of two lines has information on it. This is shown in figure 2.74. If line A has information on it, Q will be 'low'. If line B has information on it, Q will be 'high'.

SET AND RESET INPUTS BOTH TAKEN 'HIGH'.

In figure 2.75, if R and S are taken 'high' at the same time, Q and $\bar{Q}$ will go 'low'. This state is called *'UNDETERMINED'* and is avoided, if Set and Reset are then taken 'low' at the same time; one output will go 'high' and the other output will go 'low', Which output goes 'high' is random, hence the name 'undetermined'.

Fig. 2.74 A practical application.

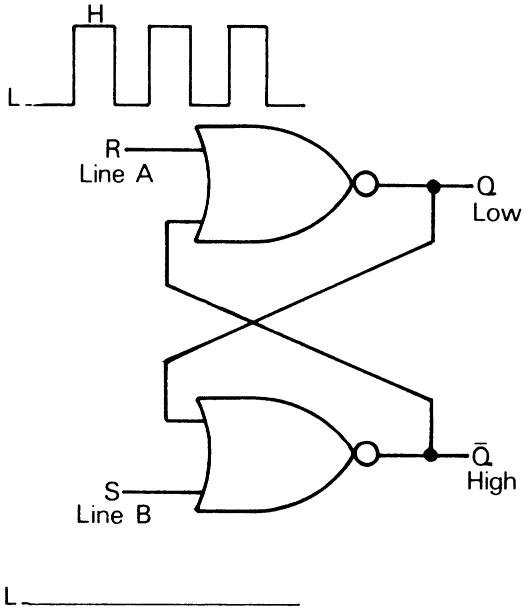
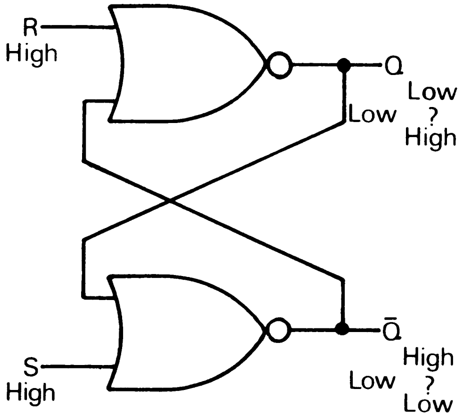


Fig. 2.75 Both Set and Reset inputs 'high'. The 'undetermined' state.



Part 3

Counters and Shift registers

Binary numbers

Our normal numbering system is called the decimal (tens) system and uses ten digits (0 – 9).

Consider the decimal number 1642 (figure 3.1). The digits are placed in their columns to show what they represent, i.e. 1 thousand, 6 hundreds, 4 tens and 2 units.

The least significant part of the number (2 units) is in the right-hand column and the most significant digit (1 thousand) is in the left-hand column.

Moving from right to left, each column represents parts of the number ten times greater than the previous column, e.g. a 3 in the tens column is ten times greater than a 3 in the units column and a 3 in the hundreds column is ten times greater than a 3 in the tens column and so on.

Fig. 3.1 Breakdown of the number 1642 in the decimal numbering system.

Thousands	**Hundreds**	**Tens**	**Units**
1	6	4	2

1 thousand + 6 hundreds + 4 tens + 2 units = 1642

The binary (two) numbering system uses only two digits 0 and 1. Consider the binary number 10100, this number can be placed in the columns representing the number as in figure 3.2.

The number 10100 in the binary system is equivalent to 20 in the decimal system, as shown in figure 3.3.

Reference figure 3.2. Moving from right to left, each column represents part of the number which is two times greater than the previous column, e.g. a 1 in the twos column is two times greater than a 1 in the units column and a 1 in the fours column is two times greater than a 1 in the twos column and so on.

Using the binary numbering system, more columns are required to represent a number compared with the decimal system.

To represent 21 in the decimal system, only two columns are required as in figure 3.4

To represent 21 in the binary system, five columns are required (figure 3.5), i.e. binary 10101 = decimal 21.

Fig. 3.2 Breakdown of the number 10100 in the binary numbering system.

Sixteens	Eights	Fours	Twos	Units
1	0	1	0	0

Fig. 3.3 Comparison of the decimal and binary numbering systems.

1	0	1	0	0
|	|	|	|	|
1 x 16	0 x 8	1 x 4	0 x 2	0 x 1
16 +	0 +	4 +	0 +	0

= Decimal 20

Fig. 3.4 21 in the decimal numbering system.

Tens	Units
2	1

Fig. 3.5 21 in the binary numbering system.

Sixteens	Eights	Fours	Twos	Units
1	0	1	0	1

VOLTAGES AND BINARY NUMBERS

Combinations of 1's and 0's may be used to represent any binary number. Therefore any binary number may be represented by combinations of two voltage levels, e.g. a High voltage for 1 and a Low voltage for 0. This principle is shown in figure 3.6 where the binary number 1010 is represented by voltages.

Any electronic circuit that can give an output at two voltage levels can be used to represent binary numbers.

Such a device may be a Flip-Flop whose Q output is either high or low. The binary number 1010 has four columns so it would require four Flip-Flops to represent it. This is shown in figure 3.7 the outputs of which are showing decimal 10.

The binary number that the Flip-Flops show in figure 3.8 represents decimal 15. As all the Flip-Flop outputs are high, then four Flip-Flops can represent a maximum of decimal 15 which equals binary 1111.

If all the Flip-Flop outputs are low at 0, then the outputs are representing a binary number of 0000 which = 0. Four Flip-Flops can therefore represent any decimal number 0 – 15 (sixteen possible states).

Five Flip-Flops can represent a maximum binary count of 11111 = decimal 31. The greater the number of Flip-Flops, the greater the number that can be represented.

Fig. 3.6 Representing a binary number with voltages.

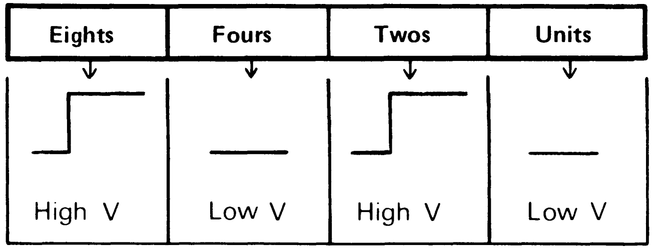


Fig. 3.7 Using Flip-Flops to represent a binary number.

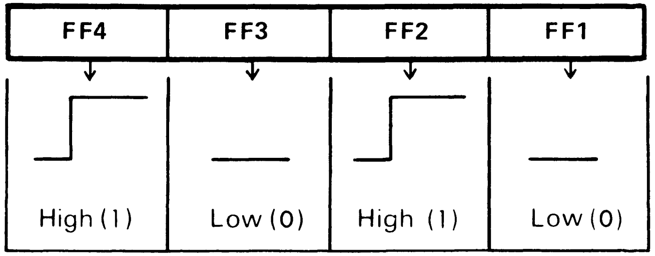
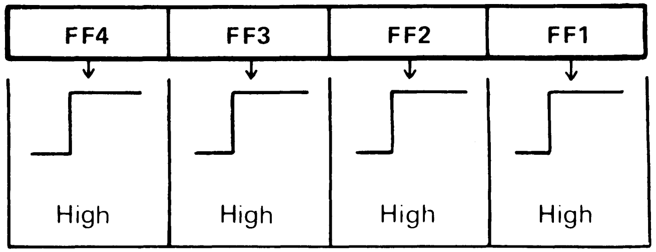


Fig. 3.8 Decimal 15 represented in binary with four Flip-Flops.



1 eight + 1 four + 1 two + 1 one = decimal 15

Figure 3.9 shows the output waveforms required from the four Flip-Flops to give a binary representation of any decimal number 0 – 15.

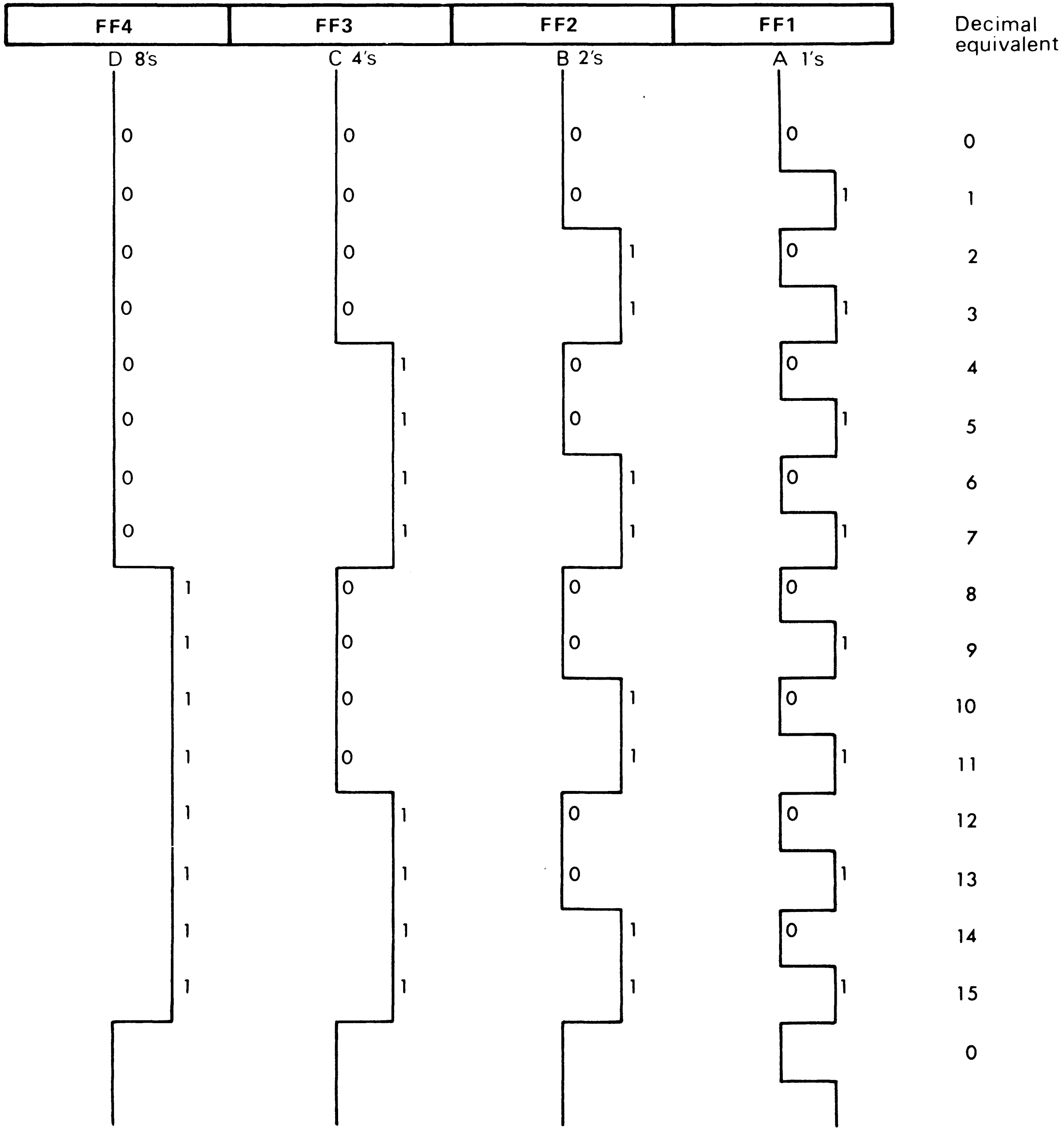
When all Flip-Flop outputs are at 0 they represent binary 0000 or decimal 0. When FF2 is high and the others low, they represent binary 0010 or decimal 2 and so on.

By convention, the 1's output is designated A, the 2's output is designated B and so on.

NOTE

FF2 output is at half the frequency of FF1 output.
FF3 output is at half the frequency of FF2 output.
FF4 output is at half the frequency of FF3 output.

Fig. 3.9 The output waveforms from four Flip-Flops for the decimal numbers 0–15.



REVISION OF THE D-TYPE FLIP-FLOP

Before considering how the Flip-Flops are connected together to form a counter, consider the action of one D-type Flip-Flop in figure 3.10.

The Q output of a D-type Flip-Flop will go to the state of the D input on the positive going edge of the clock pulse. Thus, if the D input is high, the Q output will go high as shown in figure 3.10.

The $\overline{Q}$ output is always opposite to the Q output. If the $\overline{Q}$ output is connected to the D input then the Q output will go to the $\overline{Q}$ state on every clock pulse as shown in figure 3.11.

Fig. 3.10 The action of the D-type Flip-Flop.

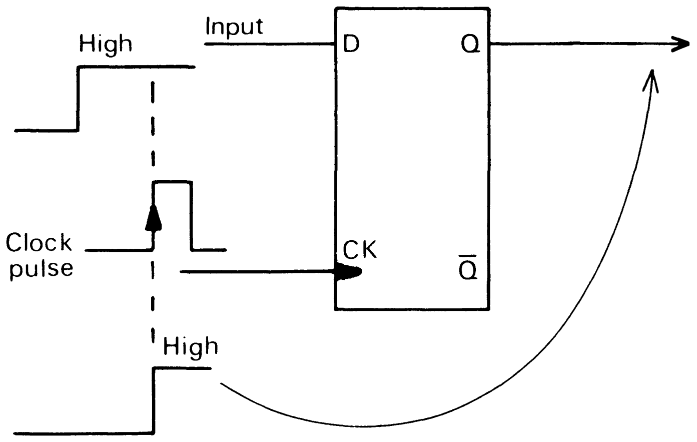
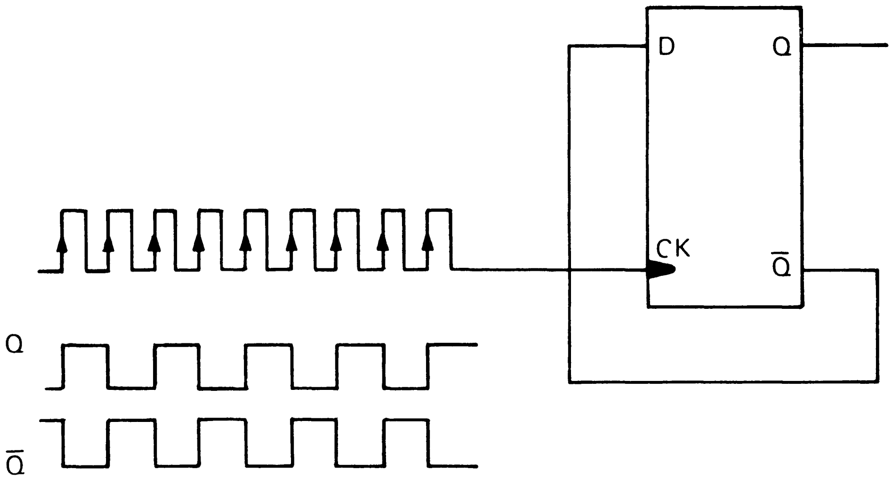


Fig. 3.11 The action of the D-type Flip-Flop with $\overline{Q}$ connected to Q.



Asynchronous counters

A counter is now constructed (as shown in figure 3.12) by connecting the $\overline{Q}$ output to the clock pulse input of the next Flip-Flop. Thus, FF1 Q output provides the clock pulse for FF2. FF2 $\overline{Q}$ output provides the clock pulse for FF3 and so on.

The output waveforms that are used for counting up are the Q outputs of each Flip-Flop.

The Q output waveform of each Flip-Flop is at half the frequency of its clock pulse input waveform.

For convenience FF1 is shown on the left where the input pulses to be counted are applied to the clock input of FF1.

At any time, the Q output waveforms will indicate how many input pulses have been fed to FF1.

Fig. 3.12 A counter using four Flip-Flops.

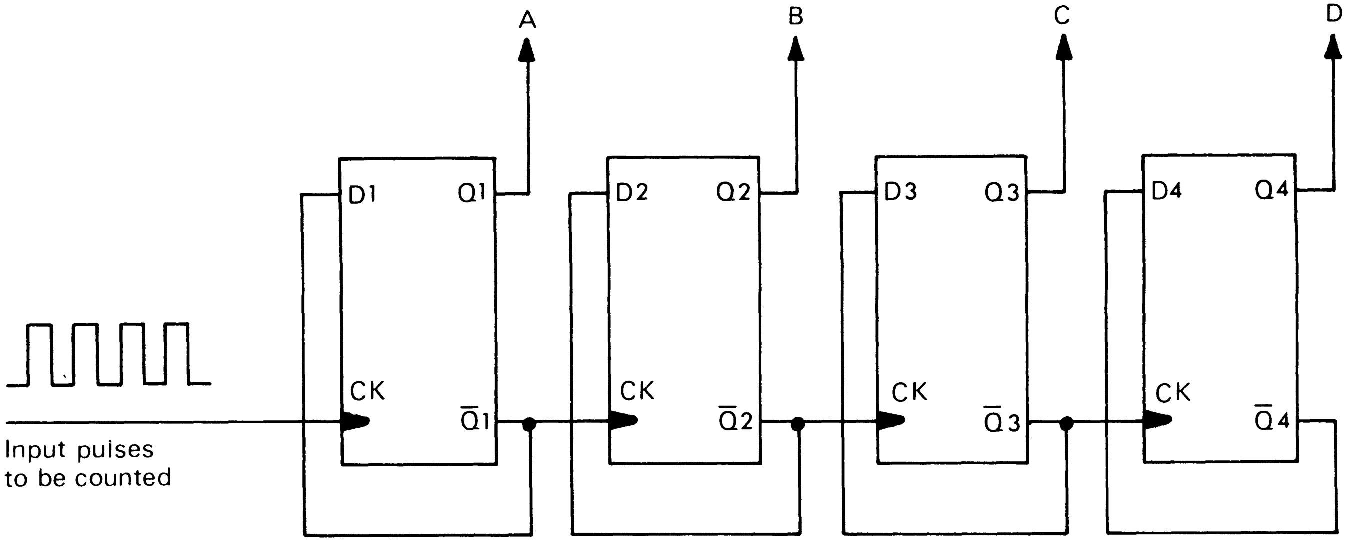


Figure 3.13 shows the $\overline{Q}$ output waveforms. Initially, all the $\overline{Q}$ outputs are at 1.

$\overline{Q}$1 output will change on the positive going edge of each clock pulse input to FF1.

$\overline{Q}$2 output will change on the positive going edge of $\overline{Q}$1 output.

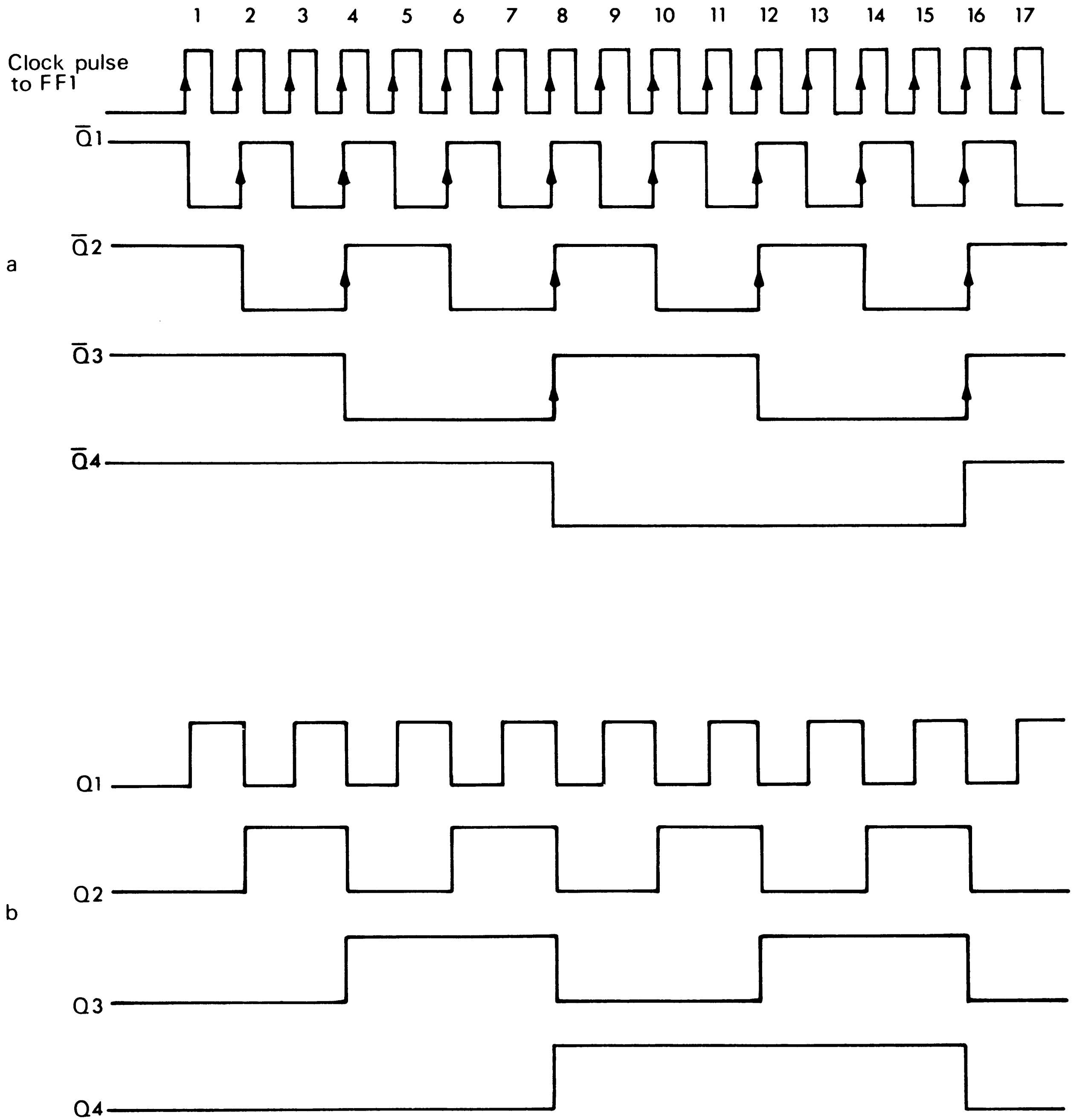
$\overline{Q}$3 output will change on the positive going edge of $\overline{Q}$2 output and $\overline{Q}$4 output will change on the positive going edge of $\overline{Q}$3 output.

Figure 13b shows the Q output waveforms which must have opposite polarity to the $\overline{Q}$ output waveforms.

It can be seen that the Q1 output changes on the positive going edge of each clock pulse to FF1 and therefore Q1 output becomes the 1's output.

Similarly, FF3 changes on every fourth clock pulse and FF4 on every eighth clock pulse to provide the 4's and 8's outputs respectively.

Fig. 3.13 The output waveforms from the counter in figure 3.12.



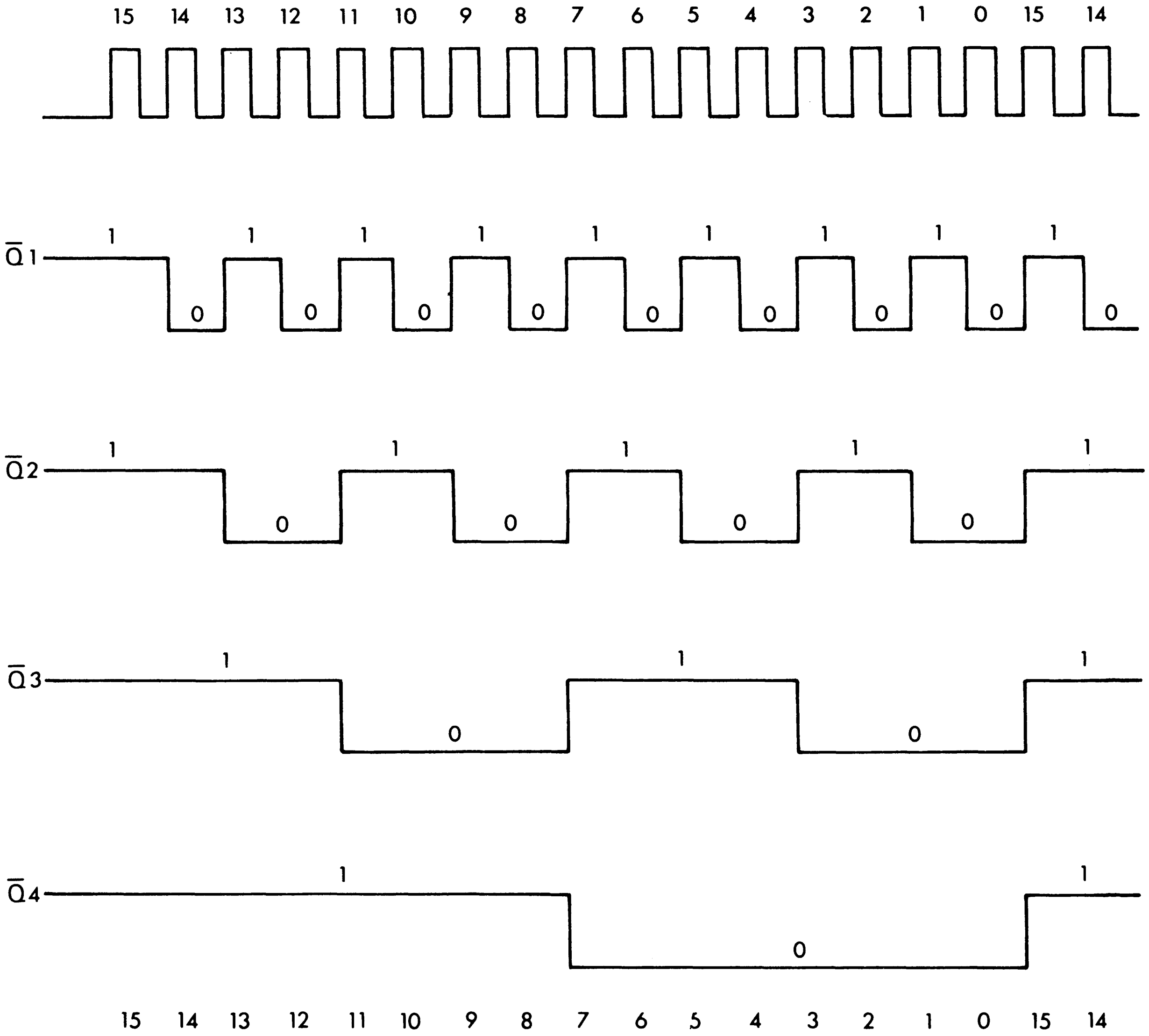
COUNT DOWN COUNTERS

Counters are also required to count down.

The counter so far described can be used to count down by taking the outputs from $\bar{Q}$ instead of Q. For convenience, the $\bar{Q}$ outputs of figure 3.13a are reproduced in figure 3.14.

Since the $\bar{Q}$ outputs are always opposite to the Q outputs, then at commencement all $\bar{Q}$ outputs are 'high', i.e. 1111 = decimal 15. On completion of the count, all Q outputs are at 0000 = decimal 0 and then return to 1111 ready to count down again.

Fig. 3.14 The waveforms when the counter is used to count down.



BLOCK SCHEMATIC REPRESENTATION OF A COUNTER

Figure 3.15 shows a block schematic of a counter.

The FF1 Q output is designated QA and represents 1's.

The FF2 Q output is designated QB and represents 2's.

The FF3 Q output is designated QC and represents 4's.

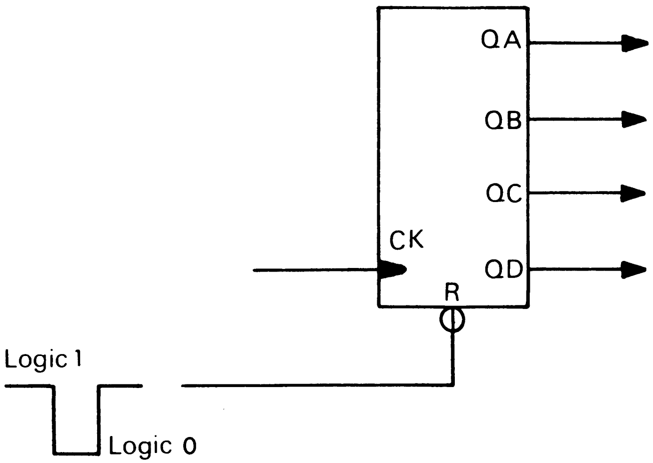
The FF4 Q output is designated QD and represents 8's.

RESETTING THE COUNTER

It is often required that counters have a master reset facility, i.e. all the Q outputs can be reset to 0 when an appropriate pulse is applied to the reset terminal.

Figure 3.15 shows a negative going reset pulse (Logic 1 to Logic 0) to reset the counter. Other counters may require a positive reset pulse (Logic 0 to Logic 1).

Fig. 3.15 A block schematic symbol of a counter.



THE COUNTER AS A DIVIDER

The output waveform frequency from any one Flip-Flop of the counter is half its input frequency. The counter can be used as a device that divides an input frequency, as shown in figure 3.16.

The input frequency to FF1 is 20kHz and the output frequency from FF1 is 10kHz. The input frequency has been divided by two.

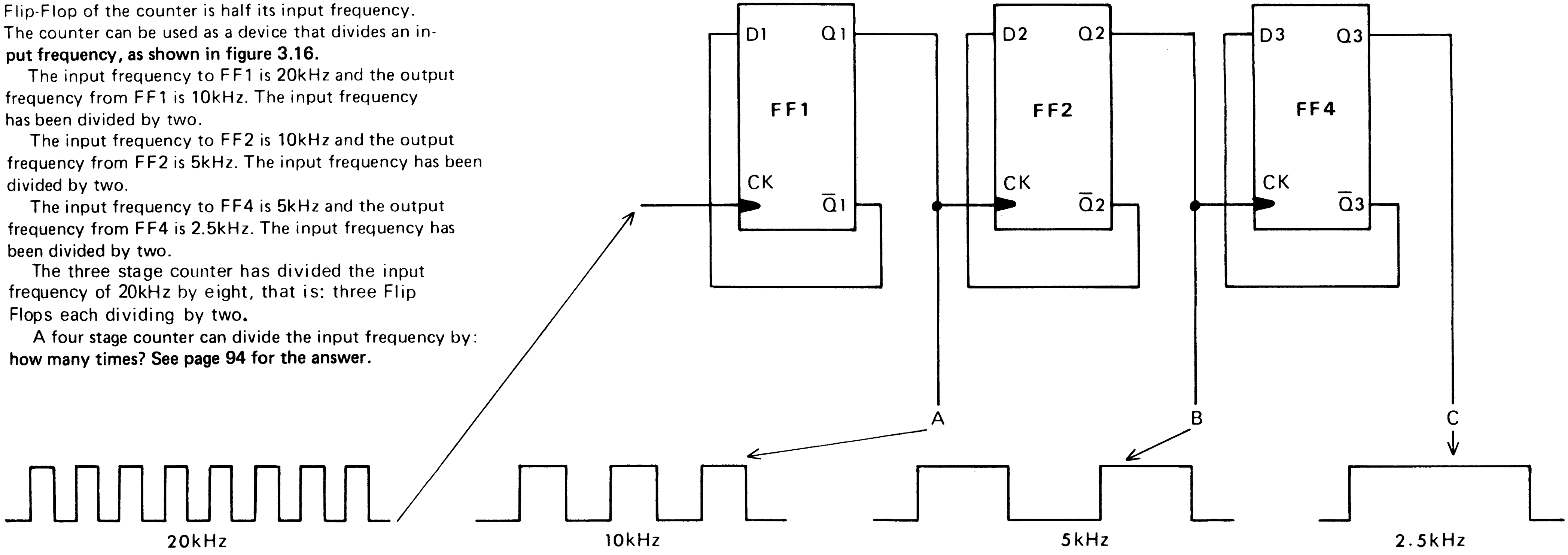
The input frequency to FF2 is 10kHz and the output frequency from FF2 is 5kHz. The input frequency has been divided by two.

The input frequency to FF4 is 5kHz and the output frequency from FF4 is 2.5kHz. The input frequency has been divided by two.

The three stage counter has divided the input frequency of 20kHz by eight, that is: three Flip Flops each dividing by two.

A four stage counter can divide the input frequency by: how many times? See page 94 for the answer.

Fig. 3.16 Using the counter as a divider.

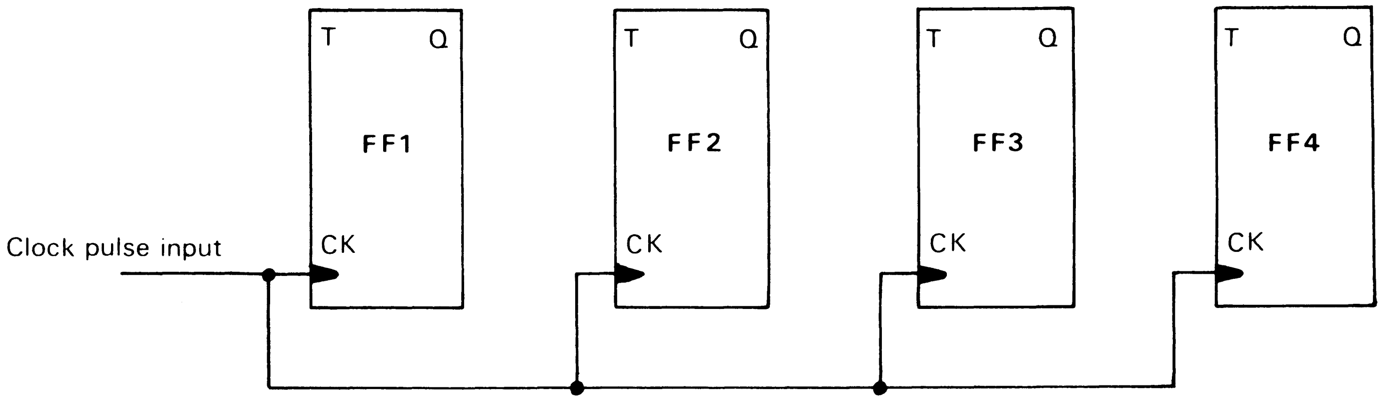


Synchronous counters

The counter so far described is called an 'asynchronous' or 'ripple' counter, i.e. each Flip-Flop (except for the first) obtains its clock pulse from the previous Flip-Flop. They are used in low frequency applications.

For high frequency applications synchronous type counters are used, i.e. each Flip-Flop has the same clock pulse as shown in figure 3.17. They can be constructed so that they can count up or down, which is why they are used for brightness control (brightness up – brightness down) in the sophisticated remote controls for THORN EMI Ferguson television receivers.

Fig. 3.17 The synchronous counter.



THE TOGGLE FLIP-FLOP

Figure 3.19 represents a toggle Flip-Flop. When T is 1, Q changes state (toggles) on the positive going edge of each clock pulse, but when T is 0, Q will not change state.

Toggle Flip-Flops can be constructed from edge triggered D types, as shown in figure 3.19.

TOGGLE ACTION

With the T terminal at 1, gate A is disabled, gate B is enabled and the $\bar{Q}$ output is transferred via the AND gate B and the OR gate C to the D input, and the Q output will go to this input state during the positive going edge of each clock pulse. Since the $\bar{Q}$ state is always opposite to the Q state, then the Q output changes (toggles) during each clock pulse.

NON - TOGGLE ACTION

´With the T terminal at 0, gate A is enabled. Gate B is disabled. The Q state will be transferred to the D input via gate A and gate C. This means that the Q and D inputs are always the same and the Q output will not respond to the clock pulse.

Fig. 3.18 A circuit symbol for a toggle Flip-Flop.

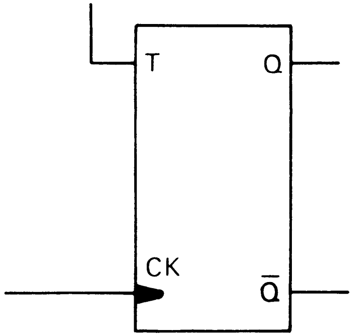
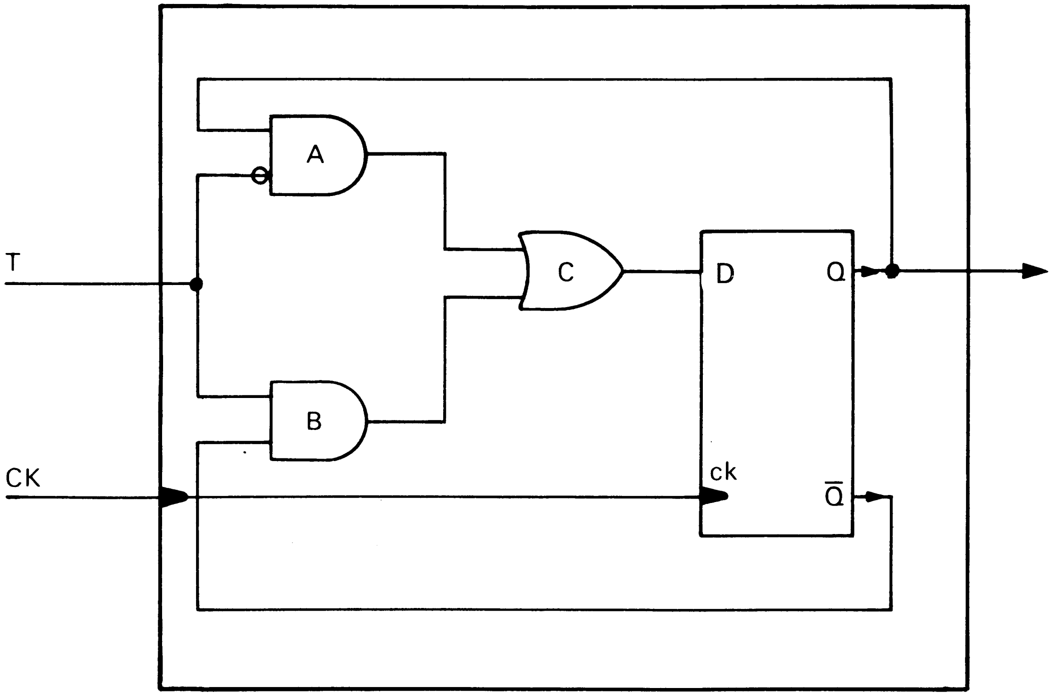


Fig. 3.19 The circuit configuration of the toggle Flip-Flop.



COUNT UP SYNCHRONOUS COUNTER

A basic four stage synchronous count-up-counter using toggle type Flip-Flops is shown in figure 3.20.

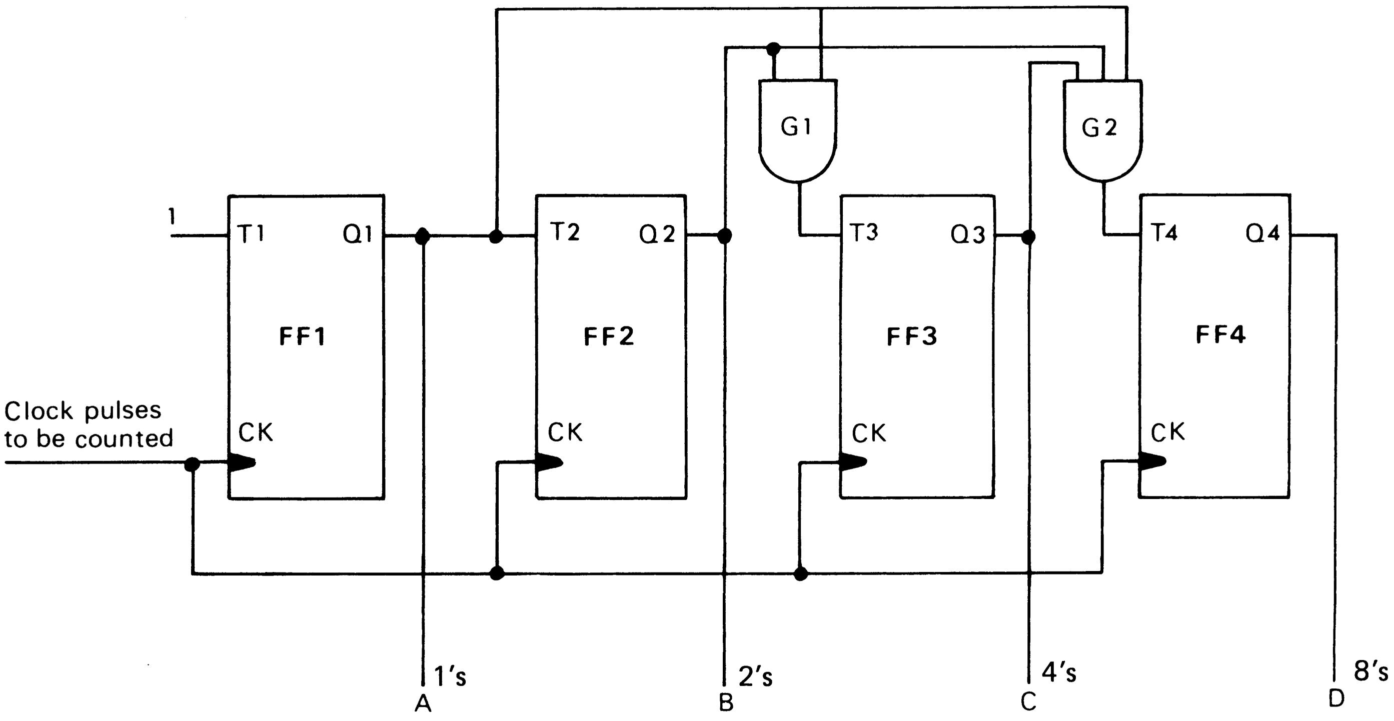
The clock pulses are fed to each Flip-Flop synchronously, i.e. at the same time.

The toggle type Flip-Flop will change state on the positive going edge of each clock pulse only when the T terminal is at 1. It will not change state if the T terminal is at 0.

In order to count up in the binary sequence, it requires two AND gates (G1 and G2). These AND gates will allow the T terminals of FF3 and FF4 to go to 1 at the correct time to change the Q state.

The explanation of this counter is given on the next page, together with its output waveforms.

Fig. 3.20 A four stage synchronous count-up-counter.



COUNT UP SYNCHRONOUS COUNTER

Initially, all Q outputs are set to 0 by a reset pulse.

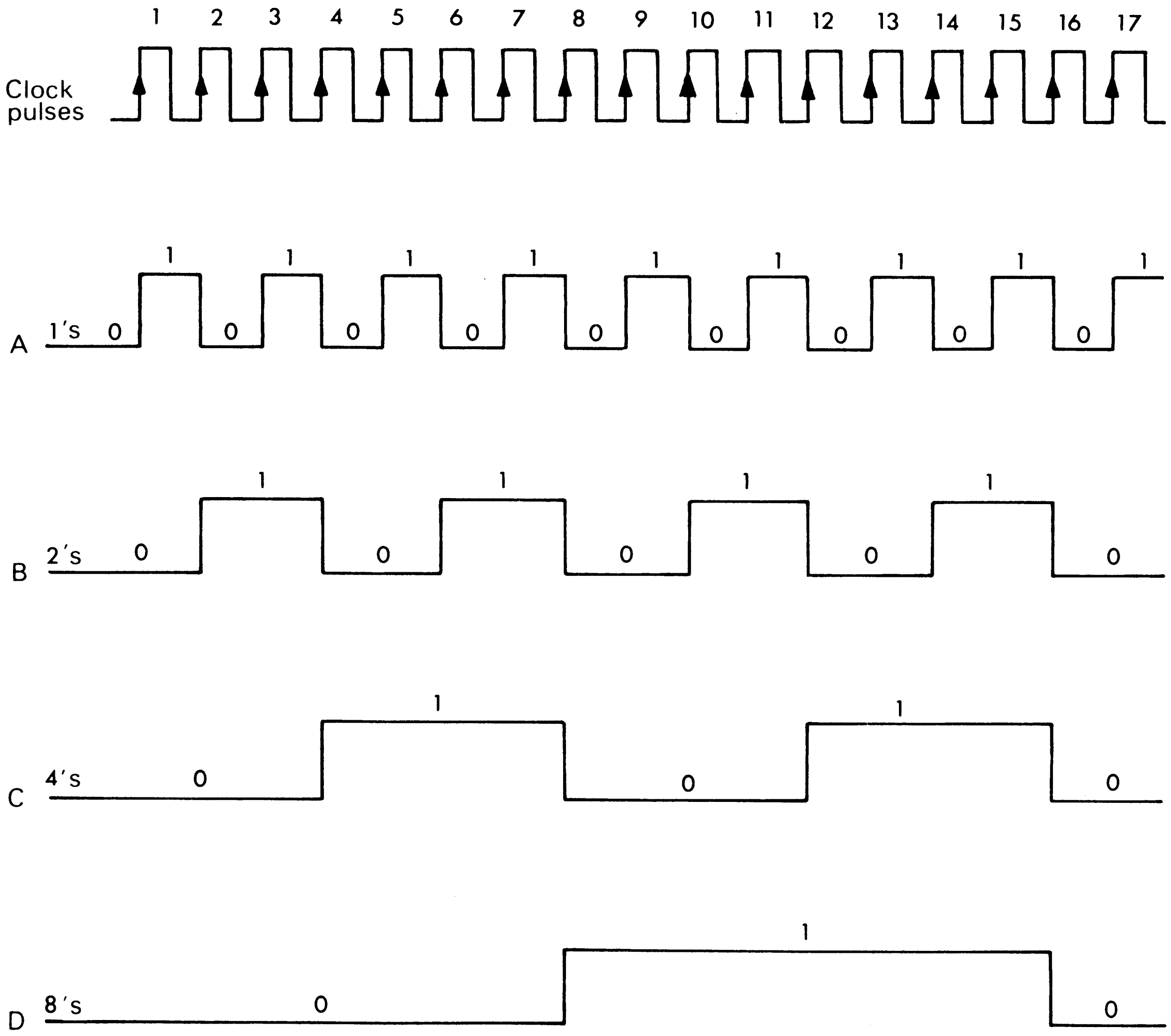
FF1 will change state on the positive going edge of every clock pulse, since T1 terminal is taken to 1. The Q1 output will therefore represent 1's in the binary count.

FF2 will only change state on the positive going edge of a clock pulse when T2 terminal is at 1 (only when Q1 output is at 1). This occurs on the positive going edge of every second clock pulse – 2, 4, 6, etc. The Q2 output will therefore represent 2's in the binary count.

FF3 will only change state on the leading edge of a clock pulse when T3 terminal is at 1. This can only be when the AND gate G1 has a 1 on both its inputs, which will be on the leading edge of every fourth clock pulse – 4, 8, 12 etc. providing a waveform to represent 4's.

FF4 will only change state on the leading edge of a clock pulse when T4 is at 1. This can only be when G2 output is at 1. In order for G2 output to be at 1 Q1, Q2 and Q3 outputs are at 1 so that G2 gate can operate. This will be on the leading edge of every eighth clock pulse – 8, 16, etc. providing a waveform to represent 8's.

Fig. 3.21 The output waveforms from the counter in figure 3.20.

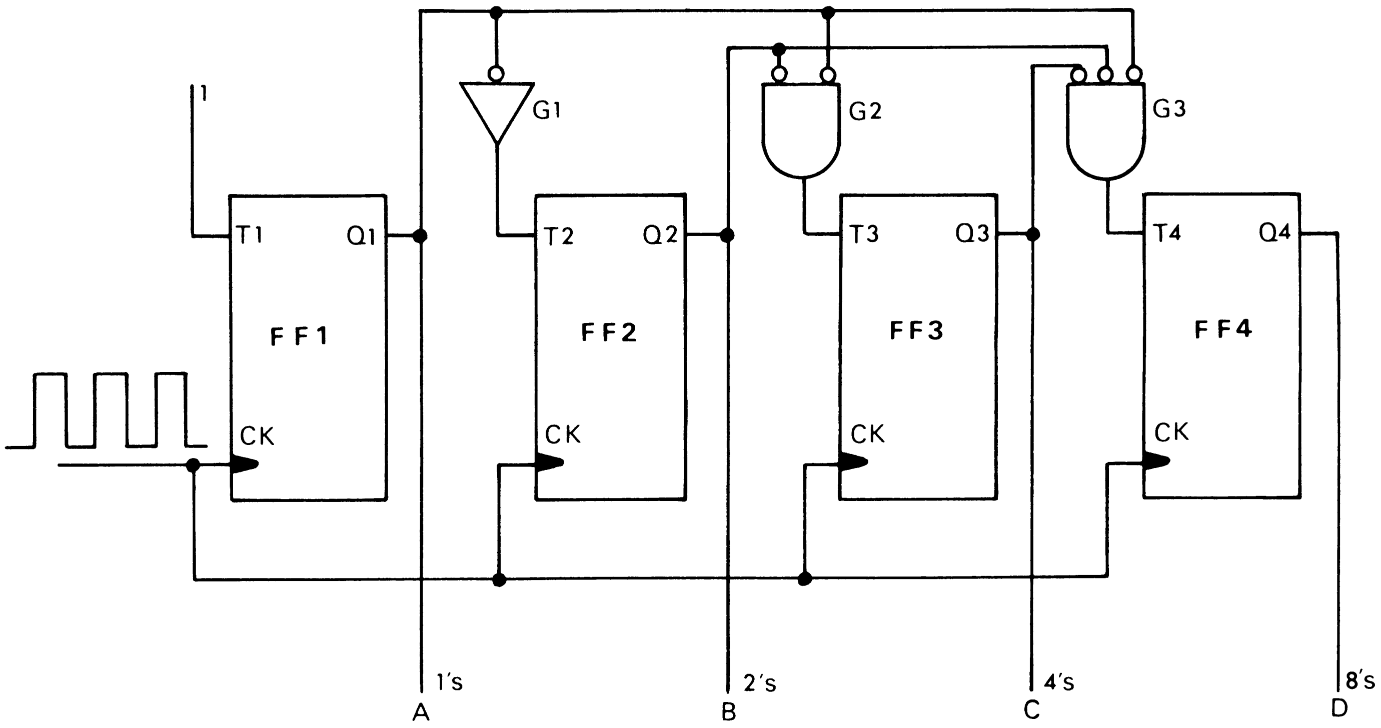


COUNT DOWN SYNCHRONOUS COUNTER

To make the synchronous counter count down, three **gates are required. G1 is an inverter. G2 and G3 are AND** gates which are activated when all of their inputs are at 0.

An explanation of this counter is given on the next page.

Fig. 3.22 A four stage synchronous count-down-counter.



Assume that initially all the Q outputs are at 1.

FF1 will change state on the positive going edge of every clock pulse since its T terminal is always at 1. The Q1 output will therefore provide a waveform to represent 1's in a binary count.

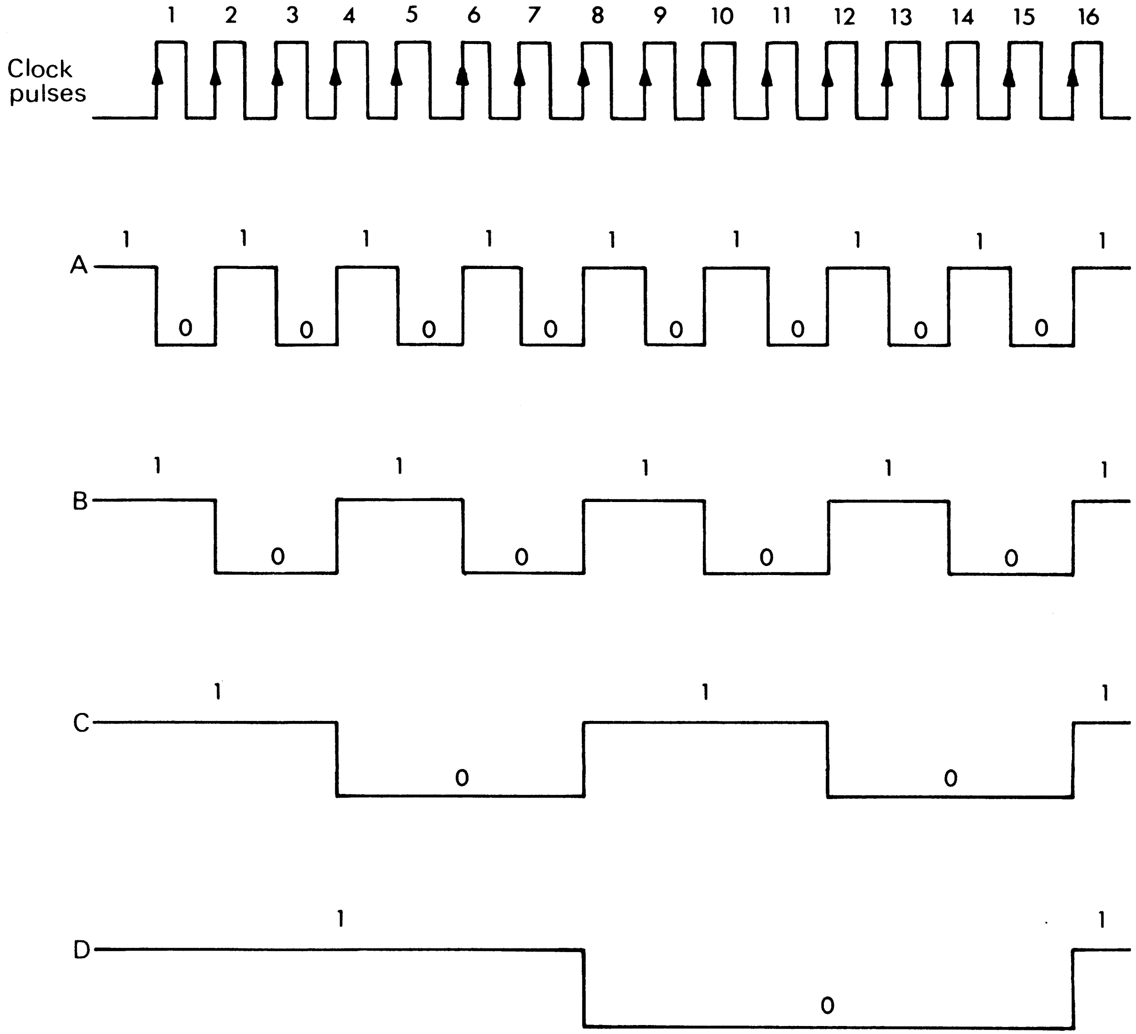
FF2 will only change state on the positive going edge of a clock pulse when its T2 terminal is at 1, i.e. when the Q1 output is 0, which is inverted to a 1 by G1. This happens on the leading edge of every second clock pulse – 2, 4, 6, 8 etc.

FF3 will only change state on the positive going edge of a clock pulse when its T3 terminal is at 1, i.e. when the Q1 and Q2 outputs are at 0 which is converted to a 1 by the AND gate G2. This happens on the leading edge of every fourth clock pulse – 4, 8, 12 etc. and the Q3 output will therefore provide a waveform to represent 4's.

Similarly, FF4 will only change state when the Q1, Q2 and Q3 outputs are at 0, which via G3 is converted to a 1 and fed to T4. This occurs on every eighth clock pulse and the Q4 output provides a waveform to represent 8's.

After the fifteenth clock pulse all Q outputs are at 0 and on the sixteenth clock pulse all Q outputs return to 1 ready to count down again.

Fig. 3.23 The output waveforms from the counter in figure 3.22.



COUNT UP/COUNT DOWN COUNTERS

By combining the logic circuitry of a count up and a count down counter together with some extra gates, a count up or count down counter can be constructed as shown in figure 3.24.

The Q outputs are used to drive the gates and the counter outputs are the $\overline{Q}$ outputs which, via inverters provide outputs of the same phase as the Q outputs.

COUNT UP

A 1 is placed on the control line and only the AND gates G1, G2 and G3 are enabled. G1, G2 and G3 will give a '1' output only when all their inputs are at 1. The '1' output passes via the OR gates G7, G8 and G9 making the T terminals 1 and the counter will count up as explained on page 80.

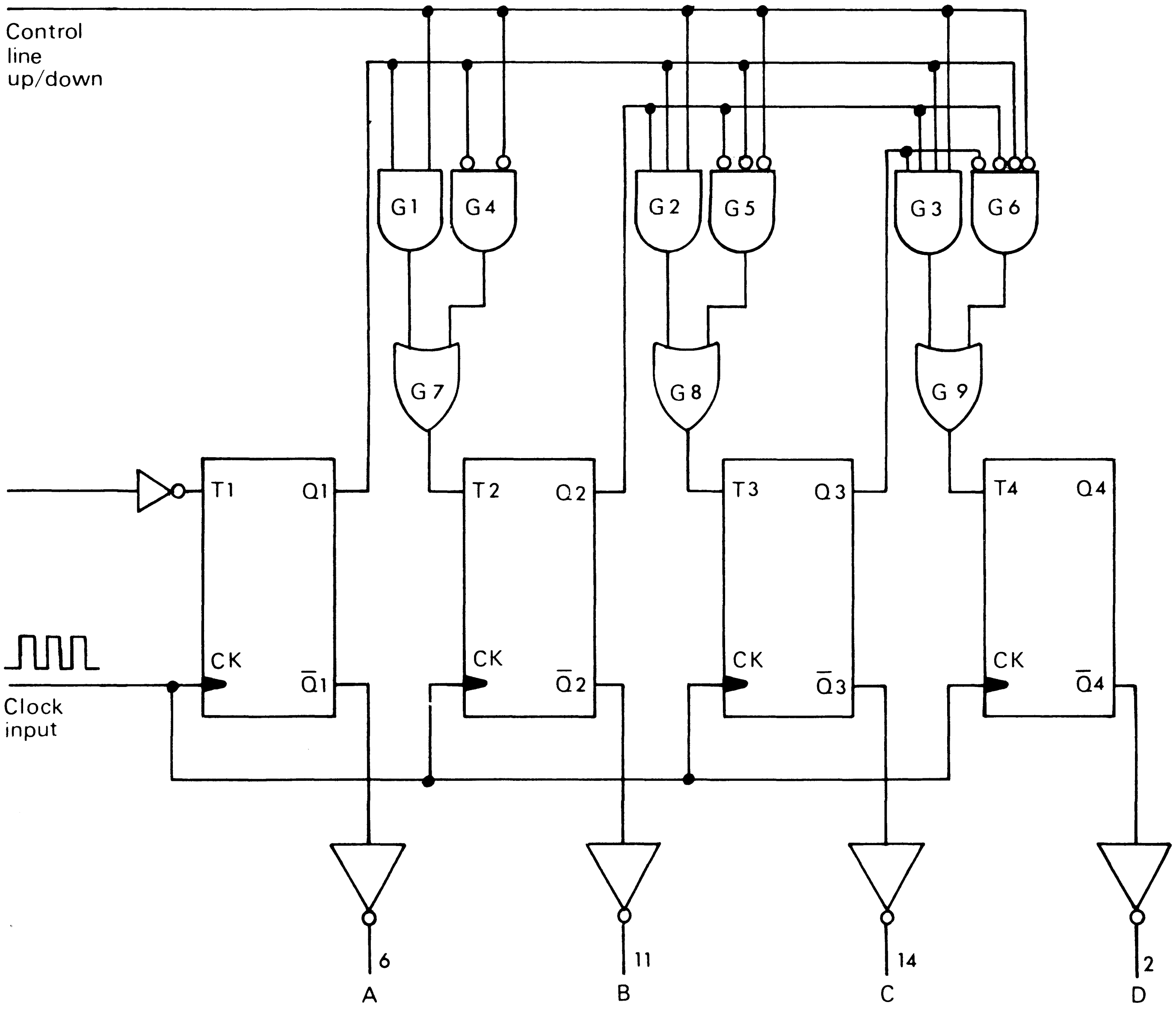
COUNT DOWN

A 0 is placed on the control line and only the AND gates G4, G5 and G6 are enabled. G4, G5 and G6 will give a '1' output only when all of their inputs are at 0, which via the OR gates G7, G8 and G9 will make the T terminals 1, and the counter will count down as explained on page 82.

Figure 3.24 is the basic count up/count down counter used for brightness control in the U703, U707, U708 and U714 remote control systems used in some THORN EMI Ferguson television receivers.

The counter outputs at pins 6, 11, 14 and 2, via a resistive network, produce a DC voltage proportional to the binary count which controls the brightness. Additional logic circuitry, not shown, stops the counter when it reaches a maximum count of 15 or a minimum count of 0.

Fig. 3.24 The count-up or count-down counter.



PARALLEL LOADING THE COUNTER

A block schematic of the counter used in the U707, U708 and U714 is shown in figure 3.25 where, as already explained, the outputs QA, QB, QC and QD are the $\overline{Q}$ outputs inverted.

The counter can be 'Preset' (parallel loaded), i.e. each Q output can be separately set to 1 or reset to 0. This facility is possible when the Preset Enable terminal (PE) is at logic 1.

Whenever the PE terminal goes to logic 1, the Q outputs take up the same condition as that on the preset input terminals, PA, PB, PC and PD. e.g. if PA is at 1, QA goes to 1 – if PB is at 0, QB goes to 0. This will happen irrespective of incoming clock pulses. Clock pulses only affect the counter when the PE terminal is at logic 0.

THE COUNTER AT SWITCH-ON

For the U707 the counter is arranged to give an output of 1001 (decimal 9) at switch-on, setting the brightness to a given level (figure 3.26).

The preset terminals are pins 3, 13, 12 and 4. Pins 3 and 4 go to +12v (logic 1) and pins 12 and 13 to Ov (logic 0). The preset terminals are therefore set to 1001. When the PE terminal goes to logic 1, the Q outputs will be forced to 1001.

The PE terminal is 'pulsed' to logic 1 at switch-on when VT27 conducts making its collector positive. As C24 charges, VT27 cuts-off and the PE terminal returns to logic 0.

With the PE terminal now at 0 the counter will respond to clock pulses and operate normally when a brightness command is transmitted.

Fig. 3.25 Block schematic of a counter used in some THORN EMI Ferguson television remote controls.

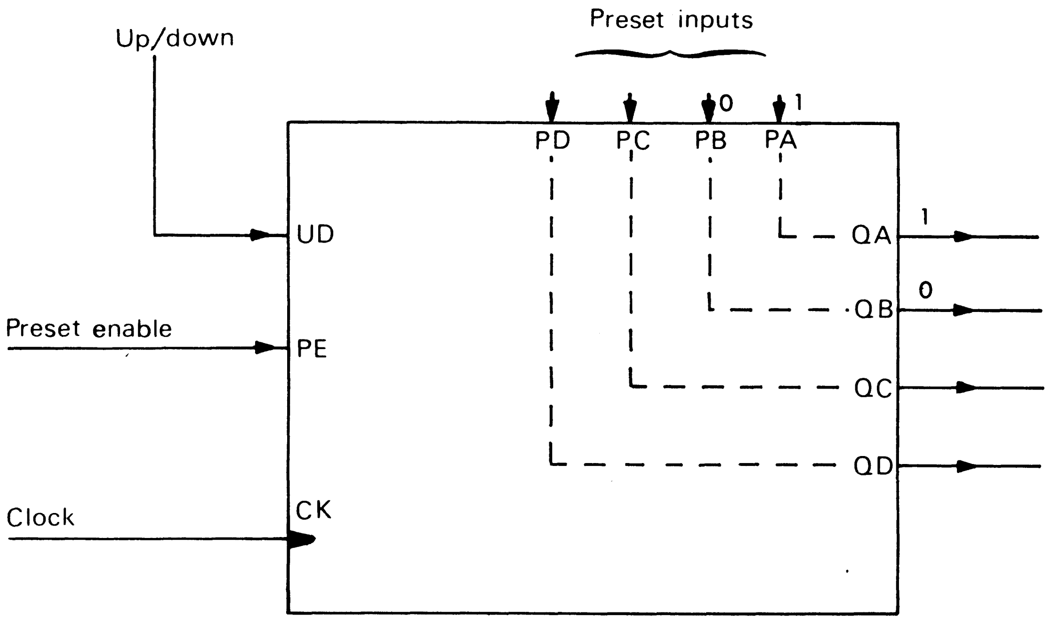
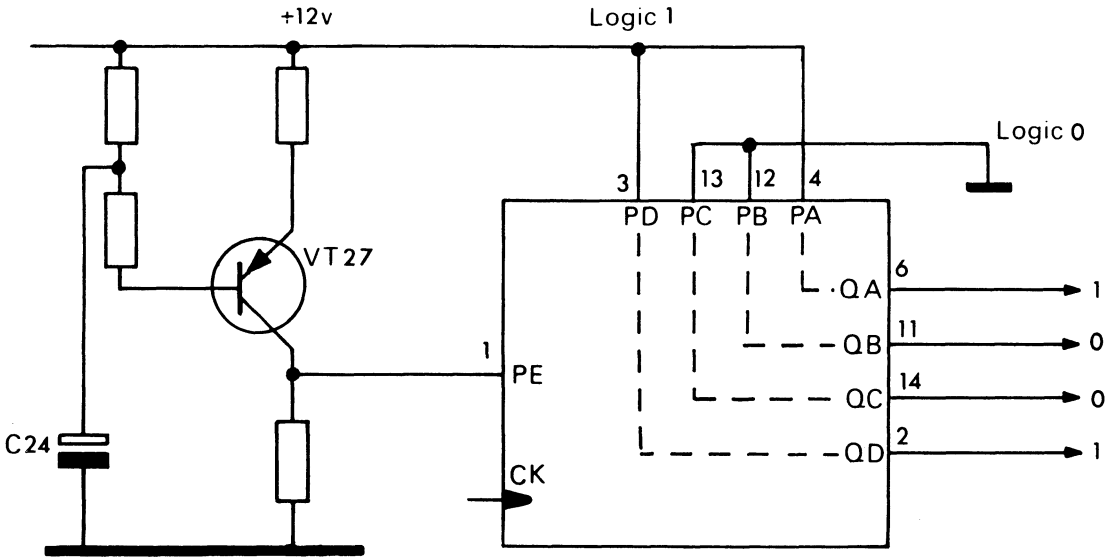


Fig. 3.26 The counter used in the brightness control circuits in the THORN EMI Ferguson U707 remote control.



PARALLEL LOADING USING THE SET AND RESET TERMINALS

Figures 3.26 and 3.27 show the PE and PA logic conditions for the QA Flip-Flop in the brightness counter in figure 3.26, the other three Flip-Flops have a similar arrangement.

NOTE

The PE terminal is common to PA, PB, PC and PD.

PE AT 0 (FIGURE 3.27)

With the PE terminal at logic 0,G2 andG3 must give a '1' output. The S and R terminals are at 1, i.e. non-activated. In this condition the PA terminal logic cannot affect the QA output and the Flip-Flop responds to its clock pulse as previously described.

PE AT 1 (FIGURE 3.28)

Taking the PE terminal to logic 1 enablesG2 andG3 to be activated. If PA is at logic 1 thenG2 is activated and its output goes to 0, this performs two functions: It activates the set terminal of the Flip-Flop, setting QA to 1; It inhibits G3 so that the reset terminal is held at 1, removing the possibility of the indeterminate condition. Conversely, if it had been required that at switch-on QA should be 0, then PA would have been at logic 0.

NOTE

Whilst operating the S and R inputs the Flip-Flop will ignore the clock or T commands.

Fig. 3.27 The QA Flip-Flop with PE at 0.

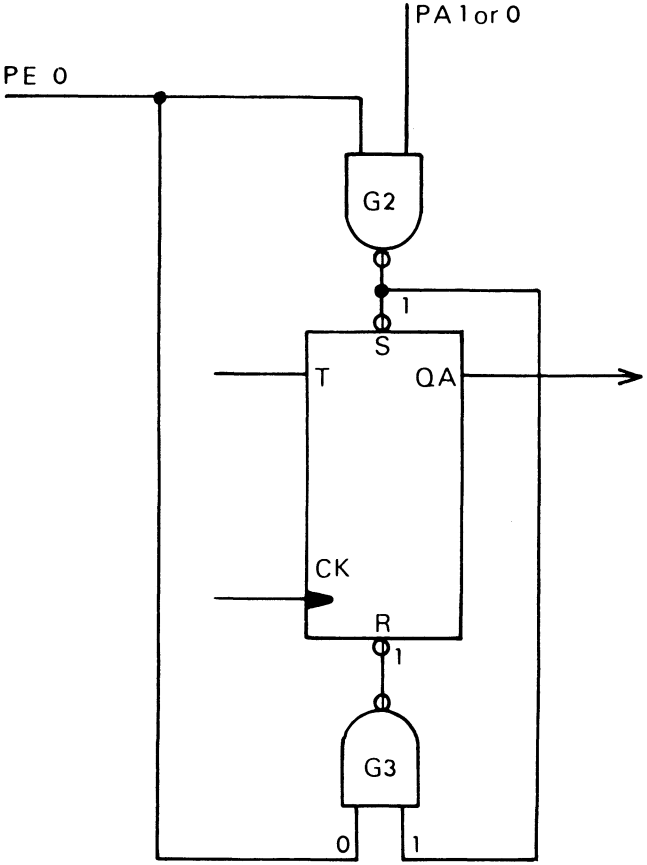
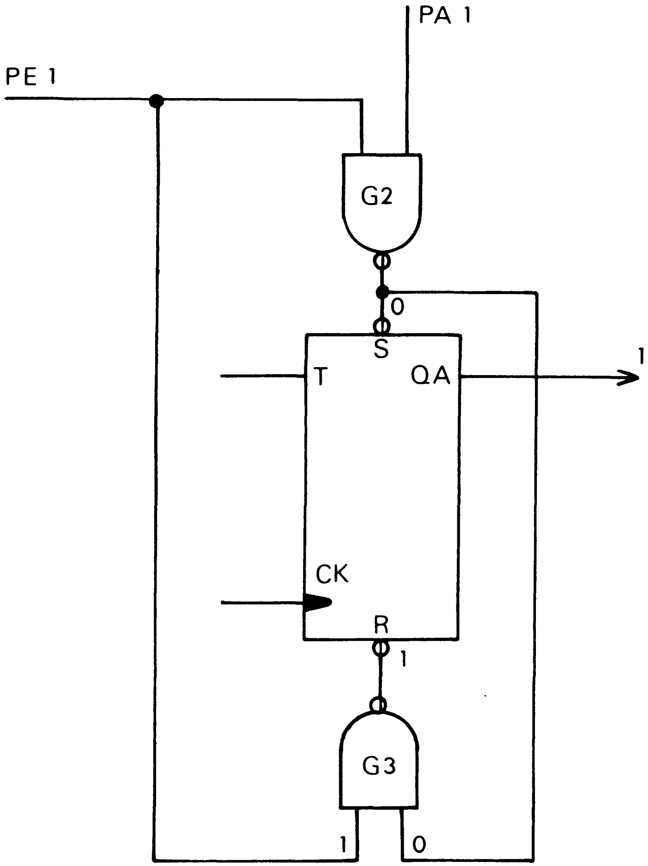


Fig. 3.28 The QA Flip-Flop with PE at 1.



Decade counters

The circuit for a decade counter using four D type **Flip-Flops and a NAND gate is shown in figure 3.29. The timing diagram for this counter is shown in figure 3.30.**

Initially, all Q outputs are reset to 0, so $\overline{Q}$ outputs are at 1. The input pulses to be counted are fed to the clock input of FF1.

The clock pulse for the other Flip-Flops are taken from the $\overline{Q}$ output of the previous Flip-Flops. This means that FF2 will change state when $\overline{Q}1$ goes to 1, i.e. when Q1 goes to 0, which is the trailing edge of Q1. Similarly, FF3 will change state when $\overline{Q}2$ goes to 1, which is the trailing edge of Q2 output.

When the counter reaches a count of:

D	C	B	A
1	0	1	0

then FF4 Q output = 1 and FF2 Q output = 1. This gives two 1 inputs to NAND gate Z taking its output low to 0, which resets the counter to zero. There is a finite time 'T' due to circuitry before the counter resets to 0.

Fig. 3.29 The decade counter.

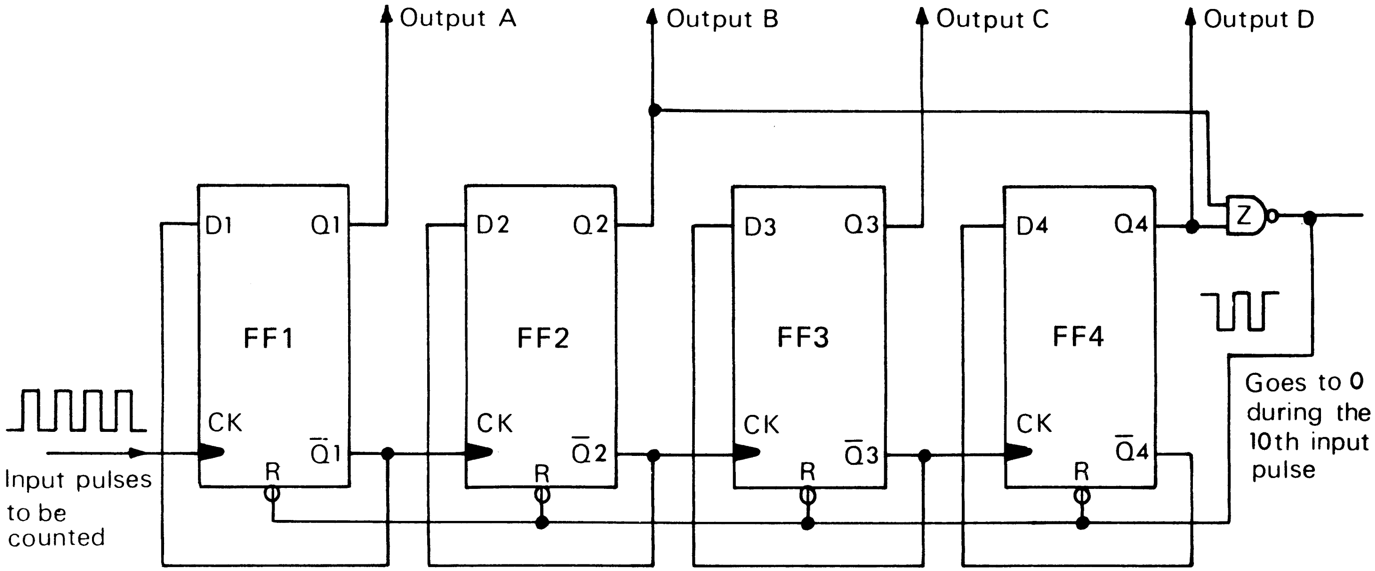
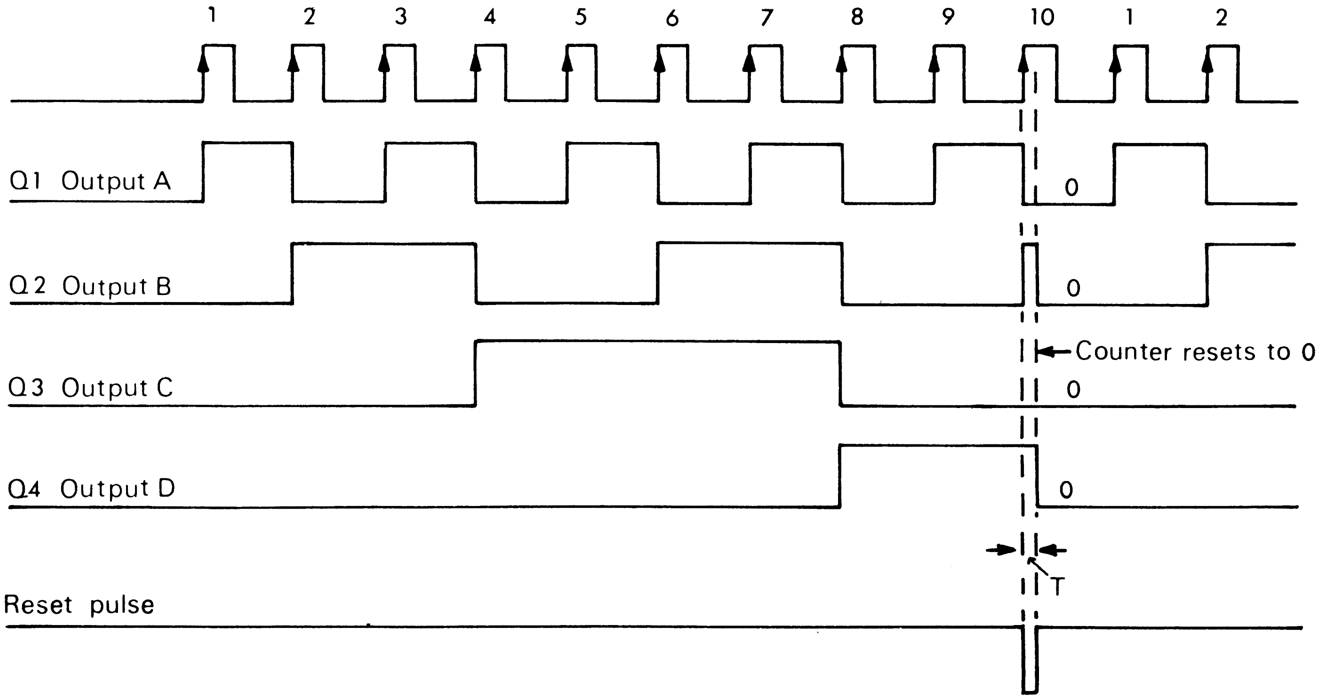


Fig. 3.30 The timing diagram of the counter in figure 3.29.



USE OF A DECADE COUNTER

A system using decade counters can be constructed to give a direct readout in decimal.

A block diagram of a system that will display hundreds, tens and units is shown in figure 3.31.

The A B C D outputs of decade counter 1, are fed into a 4 – 7 line decoder which drives a 7 segment LED display that reads from 0 – 9 giving a direct readout for units.

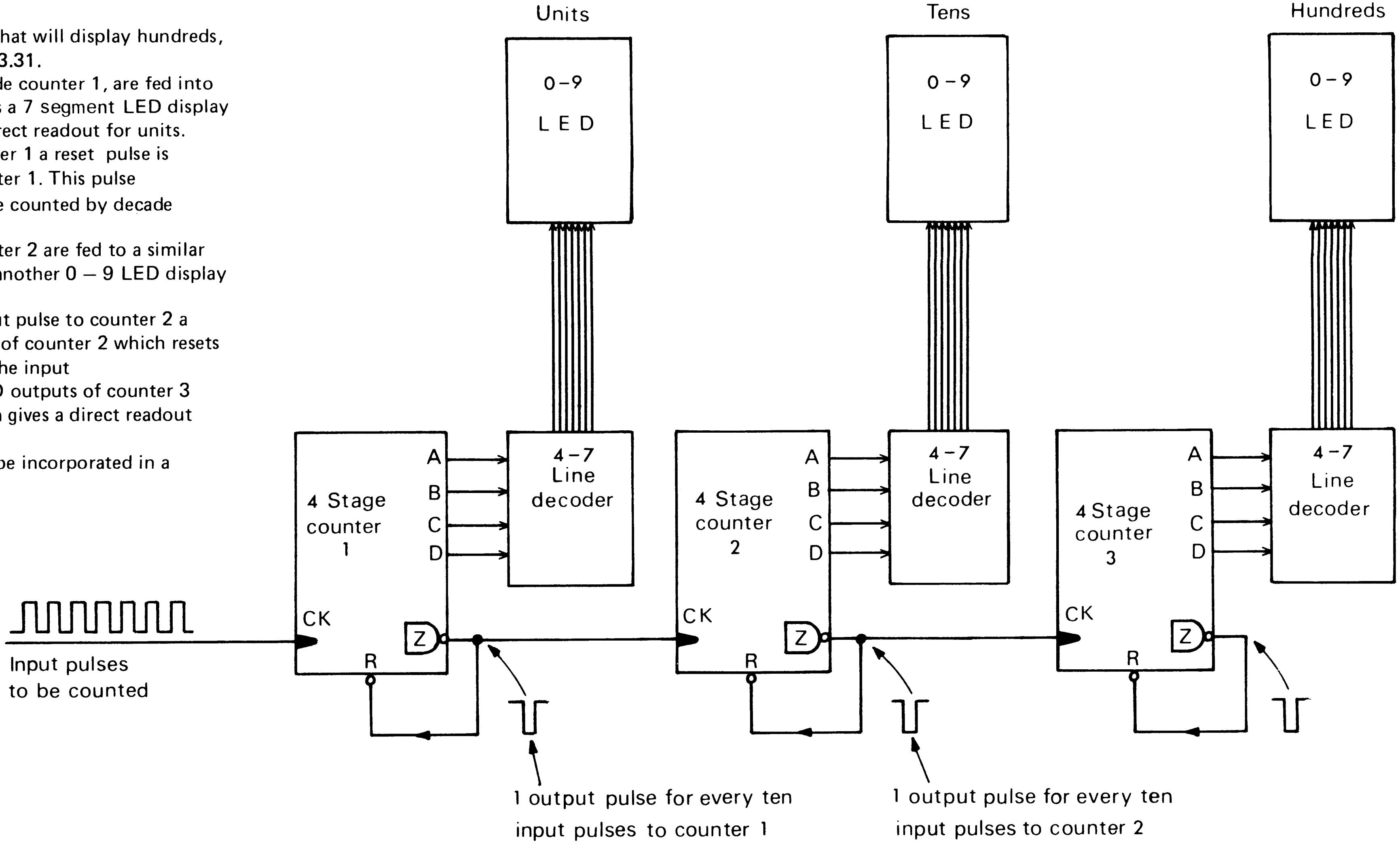
On every tenth pulse to counter 1 a reset pulse is produced at gate Z to reset counter 1. This pulse becomes the input pulse(s) to be counted by decade counter 2.

The A B C D outputs of counter 2 are fed to a similar 4 – 7 line decoder which drives another 0 – 9 LED display giving a readout for tens.

Similarly, on every tenth input pulse to counter 2 a reset pulse is produced at gate Z of counter 2 which resets counter 2. This pulse, becomes the input pulse for counter 3. The A B C D outputs of counter 3 drive another LED display which gives a direct readout for hundreds.

A system similar to this may be incorporated in a frequency meter.

Fig. 3.31 A counting system with an LED display.



Shift registers

Flip-Flops can also be connected together to form a 'Shift register'. The shift register can store 1's and 0's for as long as required. Four D-type Flip-Flops are connected to form a shift register as in figure 3.32, so that the Q output of one Flip-Flop feeds the D input of the next Flip-Flop. All Flip-Flop clock pulse inputs are connected together.

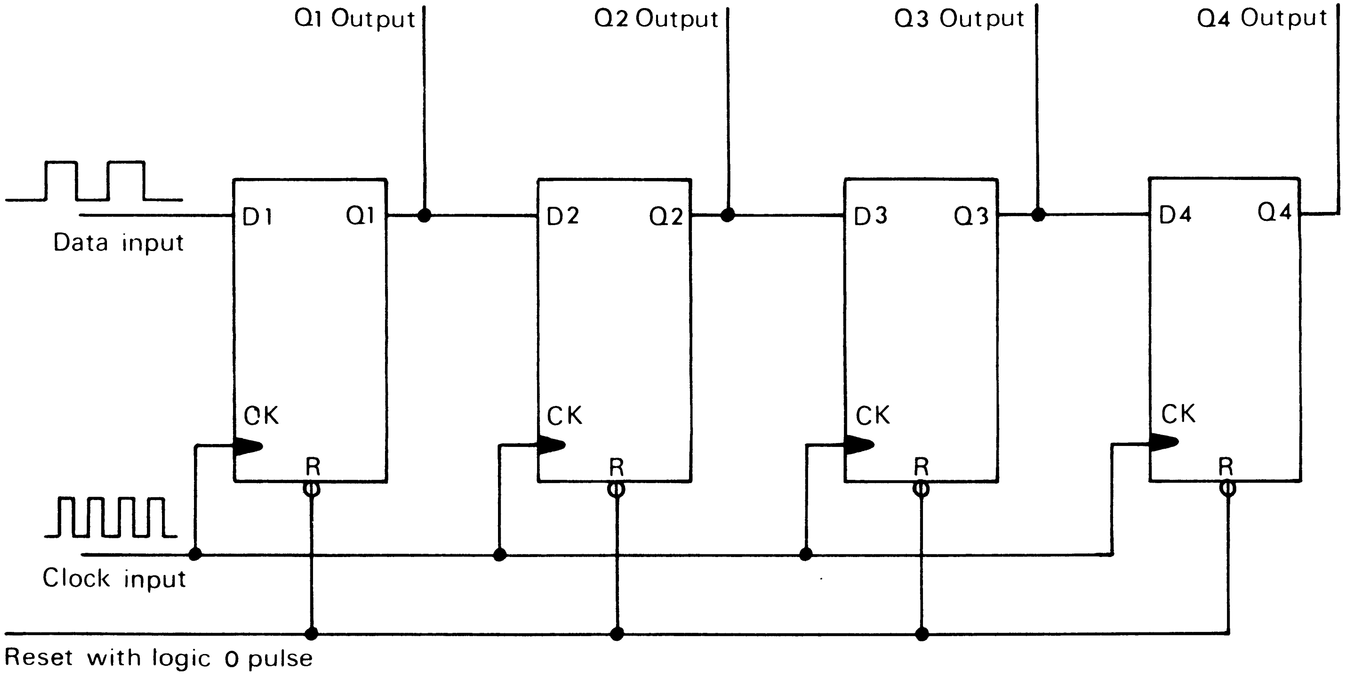
For the D-type Flip-Flop the Q output terminal will take up the condition of the D input terminal on the leading edge of the clock pulse.

Initially, all the Q outputs are set to 0. Data on the D1 terminal will be passed to Q1 on the first clock pulse. On the second clock pulse the data will pass to Q2 and so on.

Used in this way, data fed into the shift register input in serial form (one bit following another) will appear at the final Q output in the same serial form, but every bit delayed by a fixed period of time (in this case four clock pulses). This mode of operation is called: Serial In Serial Out or SISO.

Two examples of the shift register used in the SISO mode are given on pages 89 and 90.

Fig. 3.32 A shift register.



SISO WITH THE D INPUT AT 1

Initially, all Q outputs are set to 0 by a reset pulse. Suppose D1 is taken to 1 and kept there. On the leading edge of clock pulse 1, Q1 will go to 1 and remain there. Further clock pulses will not alter Q1 since D1 is now kept at 1.

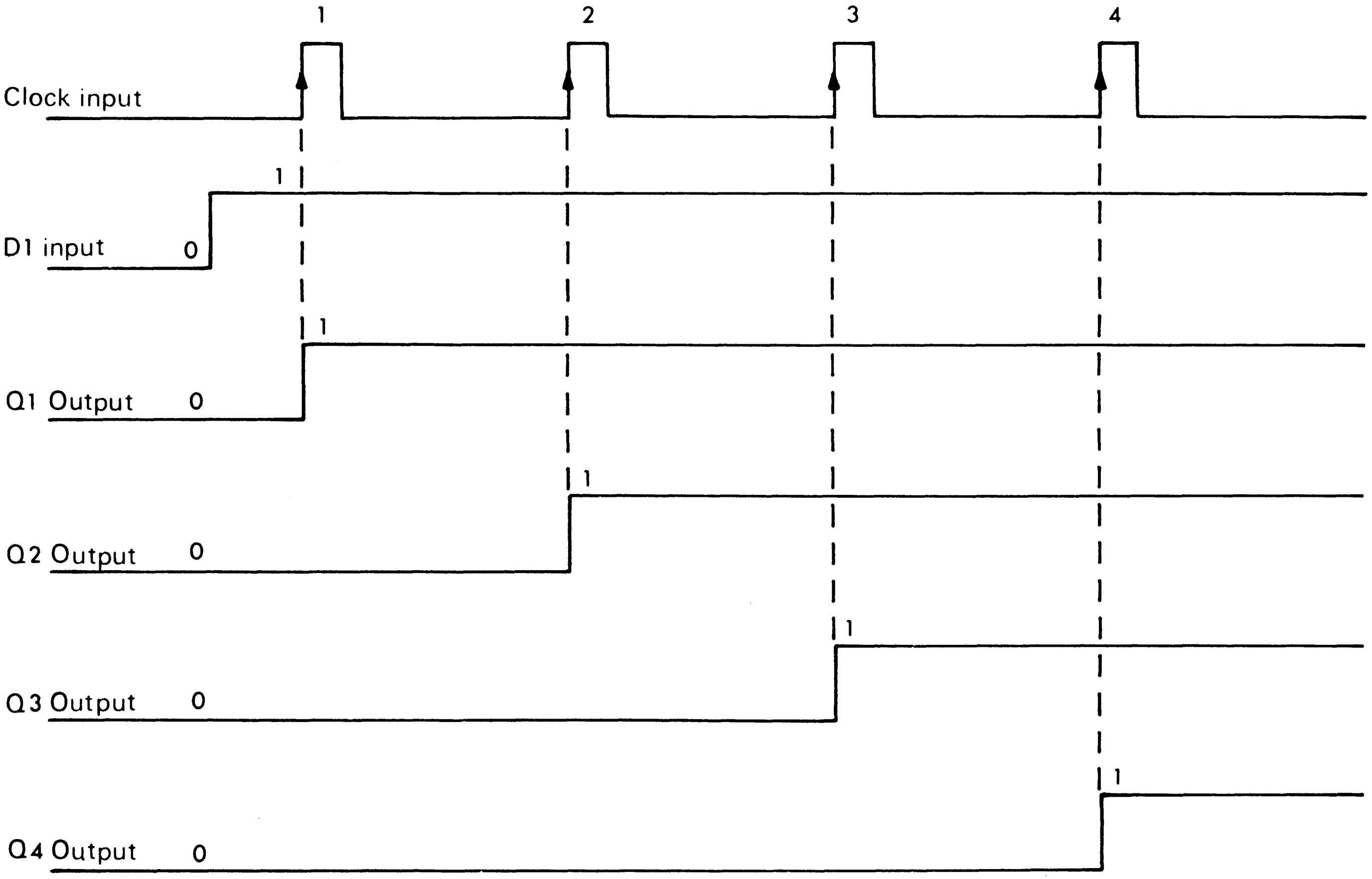
On clock pulse 2, Q2 will go to 1 because D2 (Q2) is at 1. Further clock pulses will not alter Q2 which remains at 1.

Similarly, on clock pulse 4, Q4 will go to 1 and remain there.

It has taken a period of time equal to four clock pulses to move the 1 at D1 input to Q4 output, i.e. the shift register has effectively held the 1 for four clock periods which can be regarded as a form of delay or memory.

Another way of looking at this is that the Q4 output of the shift register was '0' for a period of three clock pulses and on the fourth clock pulse the Q4 output goes to '1'.

Fig. 3.33 The timing diagram of a shift register used in the SISO mode.



SISO WITH THE D INPUT AT 0

Initially, all Q outputs are reset to 0. Now consider the action of the shift register when D1 is taken to 1 before the first clock pulse and then returned to 0 just after the first clock pulse.

NOTE

The Flip-Flop Q outputs change just after the positive going edge of the clock pulse, due to the Flip-Flop circuitry taking a finite time to respond to the clock pulse. This slight delay is in the order of a few nanoseconds (1 *nanosecond* = $\frac{1}{1{,}000{,}000{,}000}$ seconds.)

CLOCK PULSE 1

All D inputs are at 0 except D1, so Q1 becomes 1. D1 input now returns to 0 before clock pulse 2.

CLOCK PULSE 2

A few nanoseconds after the positive going edge of clock pulse 2, Q1 will return to 0 since D1 is now 0. However, because Q1 (and therefore D2) is at 1 during the positive edge of clock pulse 2, Q2 will go to 1.

CLOCK PULSE 3

Similarly, a few nanoseconds after the positive going edge of clock pulse 3, Q2 will return to 0. Because Q2 (and therefore D3) is at 1 during the positive going edge of clock pulse 3, Q3 will go to 1.

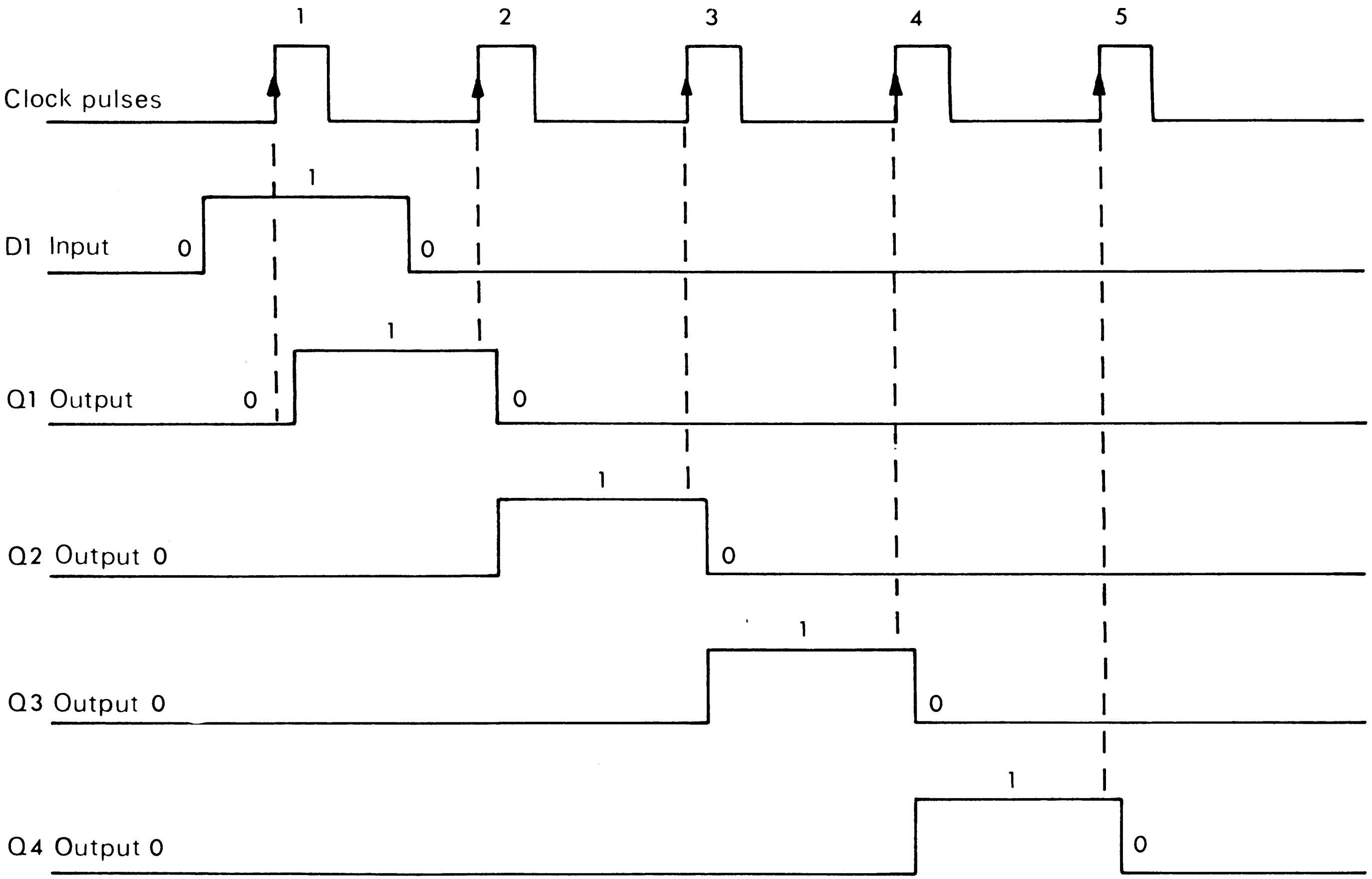
CLOCK PULSE 4

Q4 will go to 1 and all other Q outputs will be at 0.

CLOCK PULSE 5

All Q outputs will be at 0, e.g. the 1 has moved through the shift register to the final output Q4 and is removed from the register on the fifth clock pulse.

Fig. 3.34 The timing diagram of a shift register used in the SISO mode.



THE SHIFT REGISTER USED IN THE SERIAL IN PARALLEL OUT MODE (SIPO)

Suppose that a four bit word is fed into D1 in serial form, i.e. one bit following another, and that the four bit word is as follows:

1	0	1	1
first bit	second bit	third bit	fourth bit

From figure 3.35 it can be seen that each bit is fed into D1 just before a clock pulse, i.e. bit 1 is fed to D1 before clock pulse 1, bit 2 is fed to D1 before clock pulse 2 and so on.

After clock pulse 1 the first bit (1) will be at Q1 and by the fourth clock pulse will be at Q4. After clock pulse 2 the second bit (0) will be at Q1 and by the fourth clock pulse will be at Q3. After clock pulse 3 the third bit (1) will be at Q1 and by the fourth clock pulse will be at Q2. Similarly, after the fourth clock pulse the fourth bit (1) will be at Q1. The state of the Q outputs after the **fourth clock pulse is shown in figure 3.36, i.e. the four bit** word is completely in the register.

If the clock pulse is now stopped the four bit word will just stay in the shift register until it is required. The shift register is now acting as a memory.

Alternatively, the four bits can now be transferred to another part of an electronic circuit on four connections as a complete four bit word. The word was fed into the shift register bit-by-bit in serial form and is removed as a complete word in parallel form. The shift register when used like this is used in a Serial In Parallel Out mode or SIPO. This is what happens in the Teletext system. Eight bit words representing the letters to be displayed on the television screen are transmitted in serial form, accepted by an eight bit shift register, until it has the eight bits, and then transferred as a complete word to be stored in the decoder memory.

Fig. 3.35 A 4 bit word fed to the shift register.

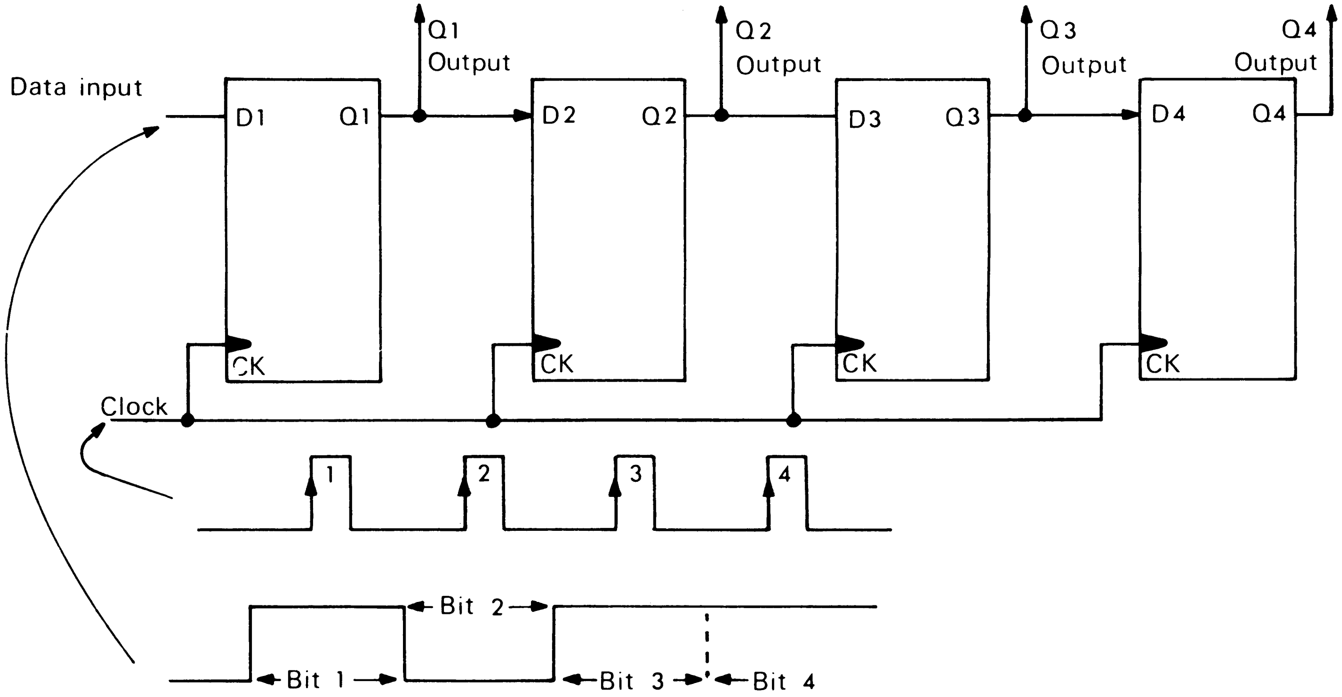
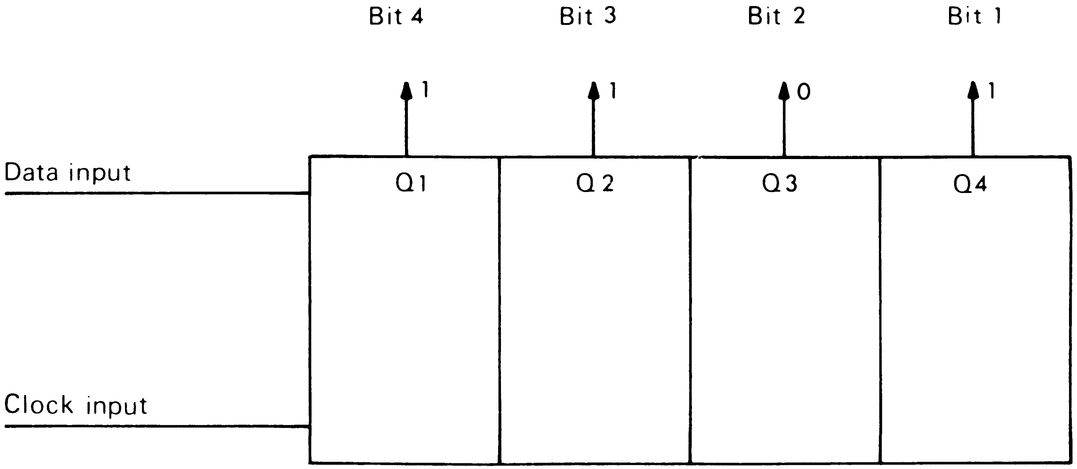


Fig. 3.36 The state of the Q outputs after the fourth clock pulse.



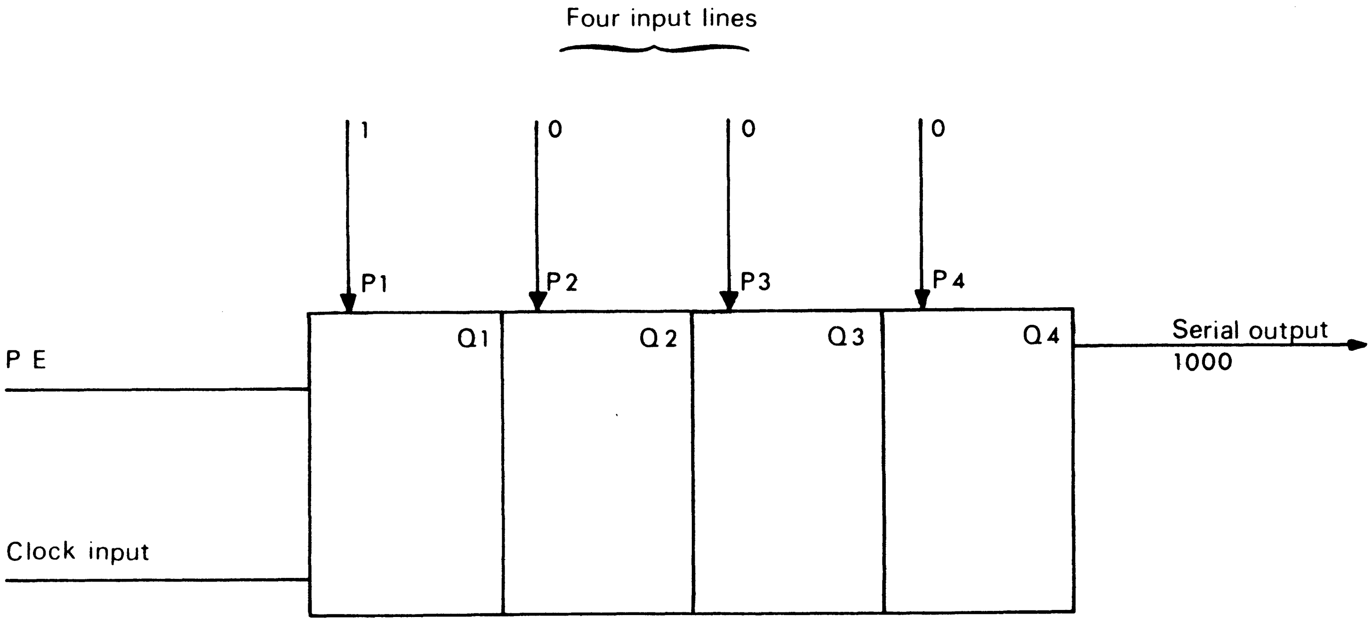
THE SHIFT REGISTER USED IN THE PARALLEL IN SERIAL OUT MODE (PISO)

A four stage shift register used in this mode is represented by figure 3.37. Four input lines carry the binary word to be loaded into the shift register. This word is loaded into the shift register by making the Q terminals of Flip-Flops go to the required 1 or 0 when the PE terminal goes to 1. The PE terminal then goes to 0 and the clock pulse moves the word through the shift register to the output terminal where it appears in serial form.

A circuit diagram of a shift register used in the PISO mode is shown on page 93 together with a circuit explanation.

This type of shift register is used in the Prestel receivers, where ten bit words in the Prestel decoder are converted to serial form, which is required for transmission over the telephone line.

Fig. 3.37 The PISO mode.



SHIFT REGISTER IN THE PISO MODE

The Q terminals of each Flip-Flop can be set to 1 or 0 by using the set and reset terminals as described below.

A 0 on the set terminal makes the Q output 1.

A 0 on the reset terminal makes the Q output 0.

With PE low, G1 to G8 outputs are high. The set and reset terminals are not activated and the shift register is in the serial mode.

Now let the PE terminal go to 1. The output of gates G1 to G8 will be as shown on the diagram.

FF1

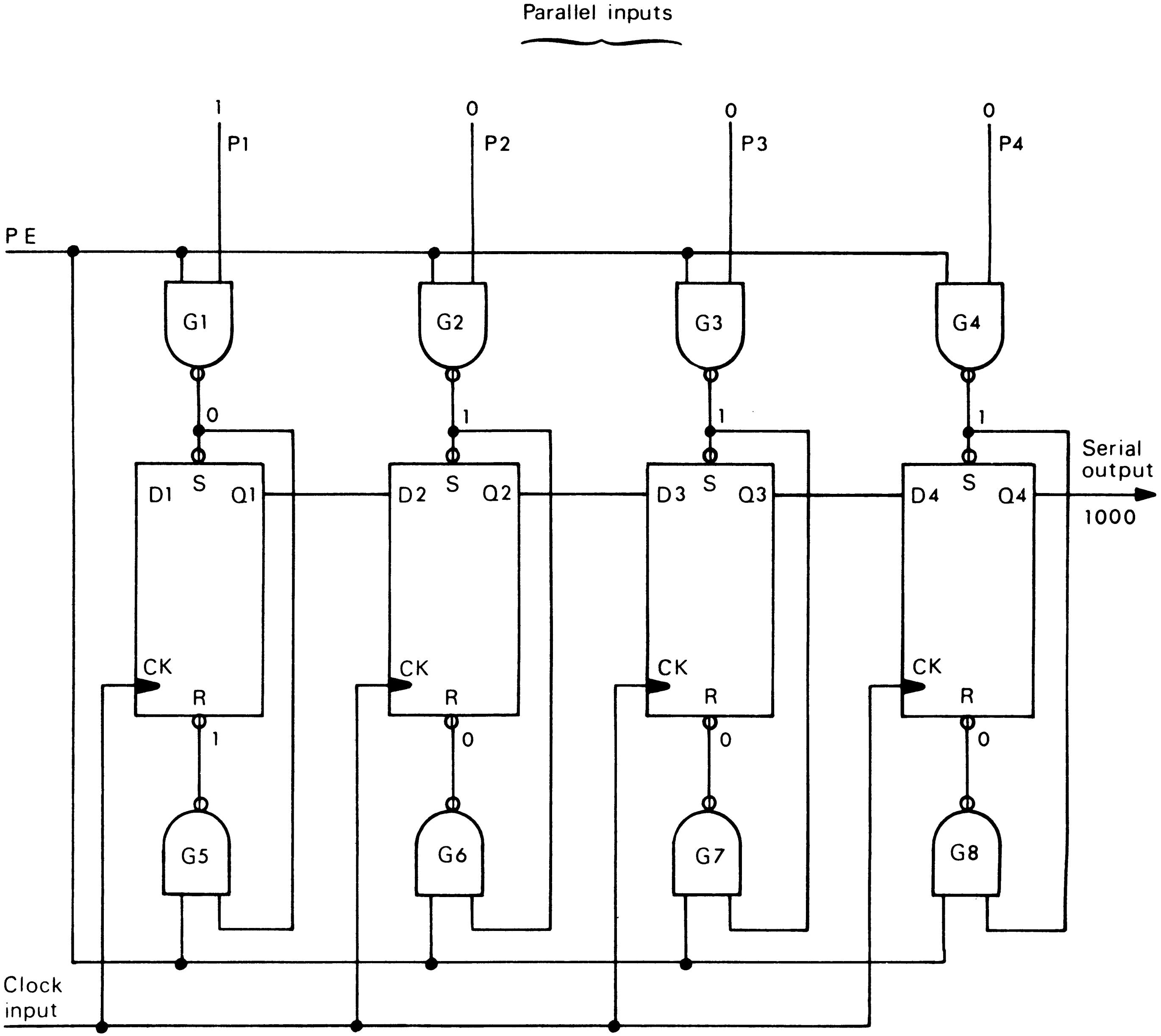
FF1's Q output will go to 1 since the set terminal is at 0 and the reset terminal terminal is at 1.

FF2, FF3 AND FF4

FF2, FF3 and FF4's Q outputs will go to 0 since the reset terminal is at 0 and the set terminal is at 1.

After loading, the PE terminal is taken to 0 and the information on the Q terminals is clocked out at Q4.

Fig. 3.38 A four stage shift register used in the PISO mode.



Provided that the PE terminal is at 1, whatever is on the G1 output will be inverted on the G5 output. Similarly, G2 output is inverted by G6 output and so on.

SISO/PISO MODE

An eight bit shift register used in both the SISO and **PISO modes, form part of the THORN EMI Ferguson U714 Teletext remote control receiver circuit.**

Its function is to inhibit channel change when transmitting numbers for Teletext pages, e.g. if page 158 is required, then when transmitting 1 the receiver must know that the 1 refers to a page number and not channel 1. This means that after the P button is pressed, channel change must not take place when the numbers 158 are pressed.

This is done quite easily by taking the emitters of the channel change transistors high whilst P158 is being transmitted.

For simplicity, a five bit shift register is shown in figure 39 to illustrate how this is achieved.

T1 is a channel change transistor. As long as its emitter is high, channel change cannot take place. Its emitter is high when G1 output is at 1.

G1 output will be 1 as long as the Q5 output is at 0. G1 output will go to 0 when the Q5 output is at 1 (two 1 inputs to G1 takes its output low).

Suppose page number P125 is required: as soon as the P button is pressed the parallel enable line goes to 1 and the P1 to P5 terminals are set as shown. The shift register will give outputs as follows:

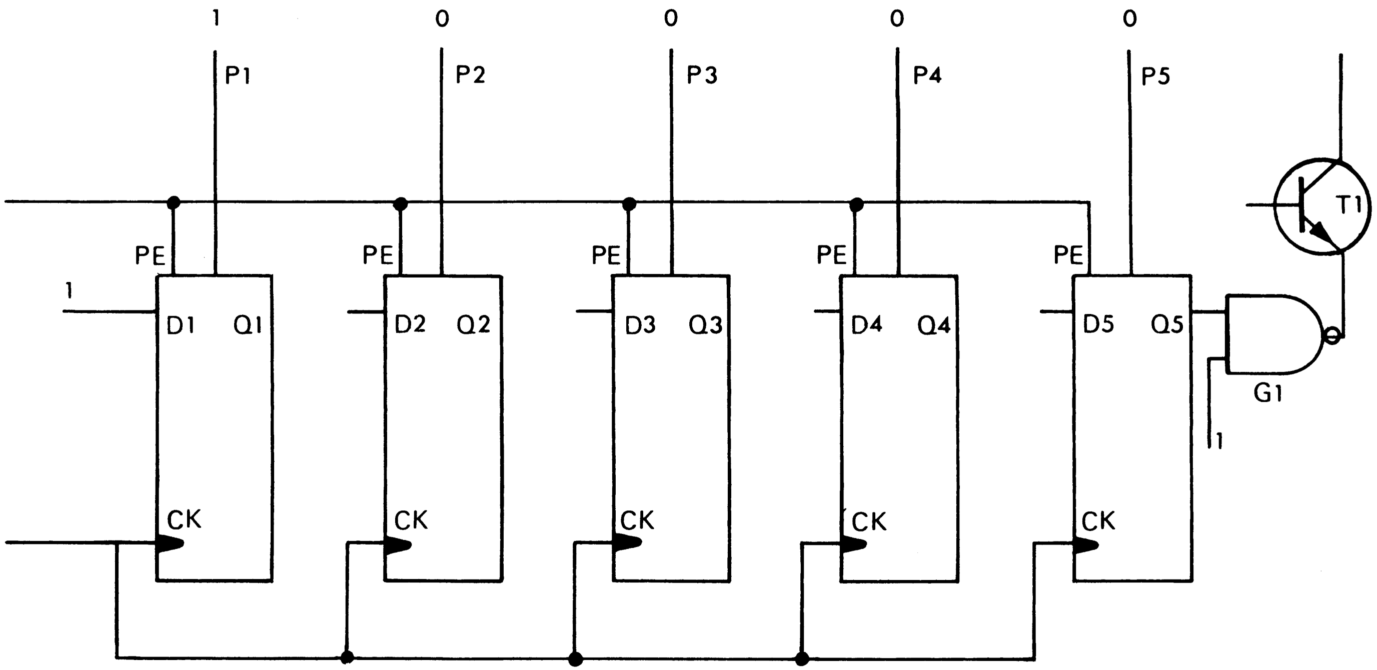
Q1	Q2	Q3	Q4	Q5
1	0	0	0	0

When the P button is released the parallel enable line goes to 0 and the shift register is now in the serial mode.

Each time a numerical button is pressed a clock pulse is generated which moves the word 10000 through the register.

So for three clock pulses (three numbers pressed for a page) the Q5 output is at 0 and T1 cannot conduct, so channel change cannot take place. After the three numbers are transmitted the Q5 output is still at 0. If another number is now called for channel change, another clock pulse will make Q5 output go to 1. G1 is activated and T1 emitter goes low and channel change takes place.

Fig. 3.39 A 5 bit shift register used in both PISO and SISO modes.



ANSWER TO THE QUESTION ON PAGE 76

A four stage counter can divide the input frequency by sixteen.

Glossary of terms

Some of the terms that follow are not covered in this book but may be used in conversation relating to logic gates.

ASYNCHRONOUS INPUTS.

Those terminals in a *FLIP-FLOP* that can affect the output state of the *FLIP-FLOP* independent of the clock. Called: Set, Preset, Reset, dc, Set and Reset, or clear.

BISTABLE ELEMENT.

Another name for *FLIP-FLOP.* A circuit in which the output has two stable states (output level 'low' or 'high'), and can be caused to go to either of these states by input signals, but remains permanently in the selected state after the input signals are removed.

CHIP (FIGURE G1)

A chip is the piece of silicon which is contained within the integrated circuit. A block of silicon is cut into slices. The slice is further cut up by scribing and breaking. This will result in hundreds of small pieces of silicon called chips. The circuit is then formed on the chip.

CLEAR.

An asynchronous input. Also called 'Reset'. It is used to restore a *FLIP-FLOP* to a 'standard' state making the Q terminal 'low'.

CLOCK.

A pulse generator that controls the timing of a system it serves, to synchronize all operations.

CLOCK INPUT.

That input on a *FLIP-FLOP* whose condition, or change of condition, controls the admission of data into the *FLIP-FLOP* through the synchronous inputs, and thereby controls the output states of the *FLIP-FLOP,* also the precise timing of any change in output state.

CML.

CML (Current Mode Logic), is a high speed logic. The transistors used in these circuits are not allowed to switch hard on (saturate). By keeping the transistors out of saturation they will operate at higher frequencies.

CMOS.

CMOS (Complementary Metal Oxide Semiconductors), is a family of logic circuits constructed with Metal Oxide coated Semi-conductor Field Effect Transistors (MOSFET's). Each gate uses both 'N channel' and 'P channel' FET's.

Typical operating conditions:

Supply voltage: 3 to 15v.

Power used: Very low indeed.

Series: The numbers used to indicate the various gates all start with a 4, i.e., 4001, 4002, 4400. For this reason it is often called the 4000 series.

COMMAND.

A pulse signal or set of signals that occur in a system which initiates a process or set of actions.

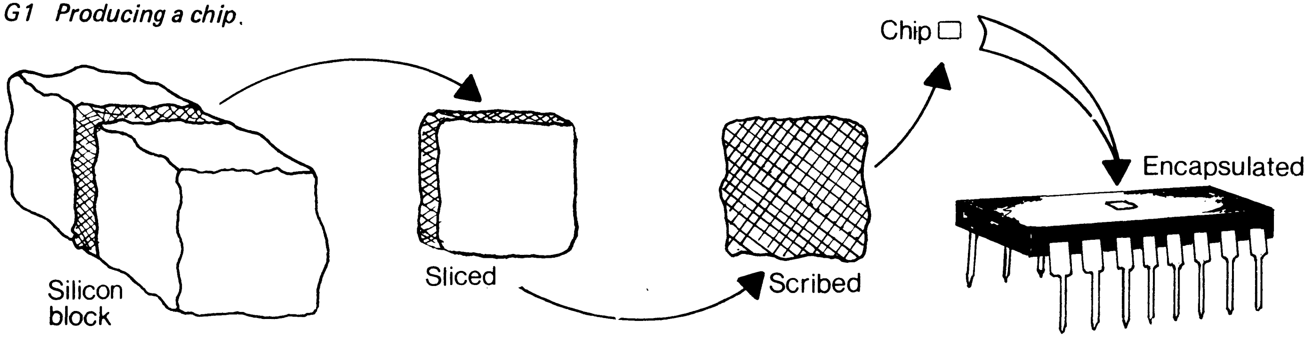


Fig. G1 Producing a chip.

COUNTER.

A device capable of changing states with the appropriate inputs. The output of the counter indicates the number of pulses that have been applied. A counter is constructed with gates and *FLIP-FLOPS.* (Counters are discussed in Part 3).

DIL (FIGURE G2).

DIL (Dual In Line), is an integrated circuit package where the lead-out pins are all in a straight line.

DTL.

DTL (Diode Transistor Logic), is a family of logic circuits where diodes are used to produce the logic function. Transistors are used only as inverting amplifiers.

ECL.

ECL (Emitter Coupled Logic), is a high speed logic. The coupling between transistors within the integrated circuit, and on the input and the output, is by emitter followers. The transistors are not allowed to switch hard on (saturate). By keeping the transistors out of saturation, they will operate at higher frequencies.

ENABLE.

An input to the enable pin of an IC will allow Data to pass through the IC to its output. The input level to the enable pin could be either a 'high' or a 'low' level, depending upon the type of IC.

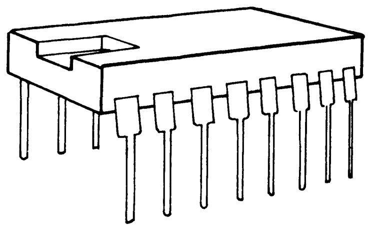


Fig. G2 A dual in line package.

FAN IN.

This is the number of inputs available into a gate or logic stage.

FAN OUT (FIGURE G3)

Fan out is the number of other gate inputs that can be fed from the output of a gate. The number is restricted, because of the current limitations in the output stage. For example an SN 7400 two-input NAND gate has a fan out of ten, which means it can feed ten inputs.

FLIP-FLOPS.

A circuit having two stable states and the capability of changing from one state to another with the application of a control signal, and remaining in that state after removal of signals. (See Bistable Element).

GATE EQUIVALENT CIRCUIT.

A diagram of basic gates, connected to perform a more complicated logic function, in order to clarify that function.

HEX.

Six gates in one package, i.e. a Hex inverter is six inverters in one IC package.

INDETERMINATE.

Used by some manufacturers instead of the term undetermined.

INHIBIT.

An input to the inhibit pin of an IC will stop the output.

The input level to the inhibit pin could be either a 'high' or a 'low' level, depending on the type of IC.

INTEGRATED CIRCUIT.

A complex or a simple circuit constructed on one piece of semiconductor material and contained in one package.

LOGIC O

The 'LOW' state is sometimes referred to as Logic O.

LOGIC 1

The 'HIGH' state is sometimes referred to as Logic 1.

LSI

LSI (Large Scale Integration) is a complete system contained in a single package. LSI could be a digital or a linear (non digital) circuit. It is a circuit that contains 100 or more gates, or circuitry of a similar complexity.

MONOLITHIC.

The complete circuit is produced on one silicon chip.

MSI.

MSI (Medium Scale Integration), is a complete system contained in a single package. MSI could be a digital or a linear (non digital) circuit. It is a circuit that contains between 12 and 100 gates or circuitry of a similar complexity.

NEGATIVE LOGIC.

The 'high' logic state can be called Logic 1. The 'low' logic state can be called logic O. However for a system with a predominance of negative going pulses, negative logic is used. This means:—

The 'high' logic state is called logic O. The 'low' logic state is called logic 1.

NMOS

NMOS ('N' type Metal Oxide Semiconductor Logic), is a logic family using Metal Oxide coated Field Effect Transistors (MOSFET's). NMOS gates are made with 'N' channel FET's.

OPEN COLLECTOR (FIGURE G4).

With an 'open collector gate' a load has to be supplied for the gate to work into. This load is often a resistor connecting the output to HT.

PMOS.

PMOS ('P' type Metal Oxide Semiconductor Logic), is a logic family which uses Metal Oxide coated Field Effect Transistors (MOSFET's). PMOS gates are made with 'P' channel FET's.

PRESET.

An input similar to the 'Set' input. It is often used to load the starting condition into a device at switch-on.

Fig. G3 Fan out.

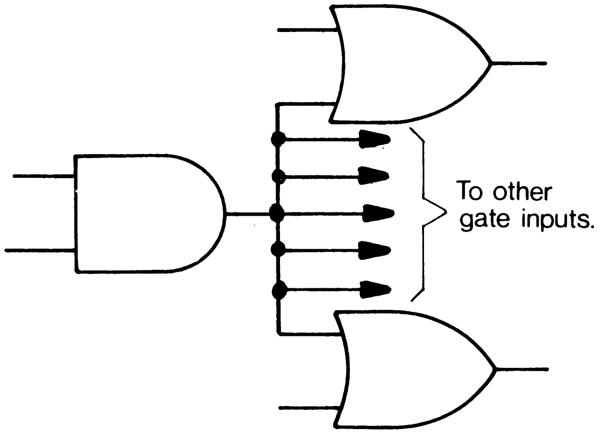
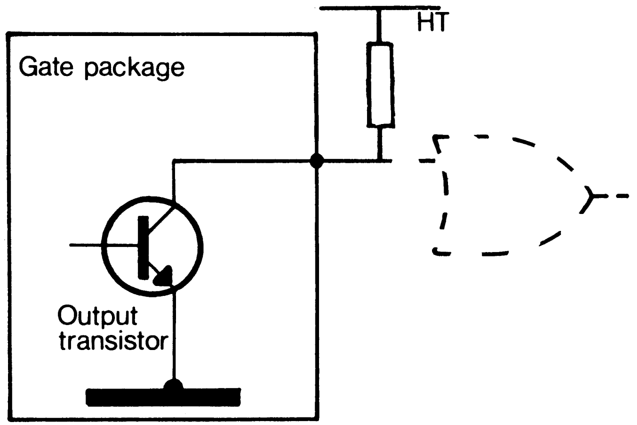


Fig. G4 Open collector.



PROPAGATION DELAY (FIGURE G5)

This is the time it takes for the input data to change the output state of the gate.

QUAD (FIGURE G6)

Four gates in one package. i.e. A. Quad two input *AND* package contains four *AND* gates.

QUIL (FIGURE G7)

QUIL (Quad in Line), is an integrated circuit package where the lead-out pins are alternately staggered.

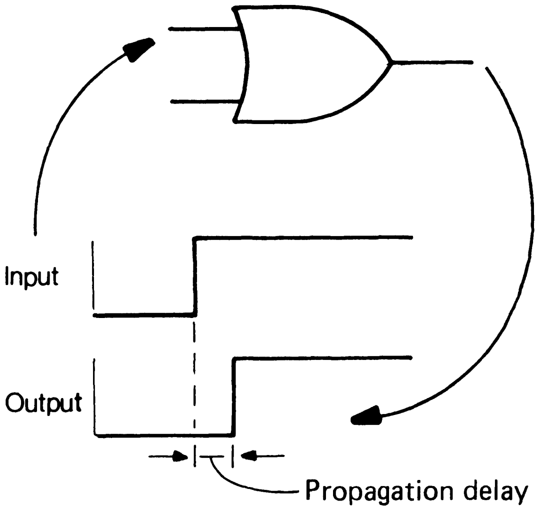
REGISTER.

A number of *FLIP-FLOPS* used to store a number of digits, i.e. a 4 digit register needs 4 *FLIP-FLOPS.*

RESET.

Also called 'clear', it is an input on a *FLIP-FLOP* not controlled by the clock, and used to affect the Q output. It is this input through which signals can be entered to make the Q output 'low'. It cannot make Q go 'high'.

Fig. G5 Propagation delay.



SCHMITT GATES (FIGURE G8).

In going from 'low' to 'high' or from 'high' to 'low' output, the gate passes through a region where it is between 'high' and 'low' states. In this region the gate can act as a high gain amplifier. In doing so, oscillations and instability can occur. The Schmitt gate makes use of Schmitt Triggers on the input to the gate. When a certain threshold voltage is reached, the gate very rapidly changes states.

SCHOTTKY.

Normal TTL is limited in speed to below 50 MHz. By the use of Schottky diodes Schottky TTL is a high speed logic that operates up to 150 MHz.

Fig. G6 A quad package.

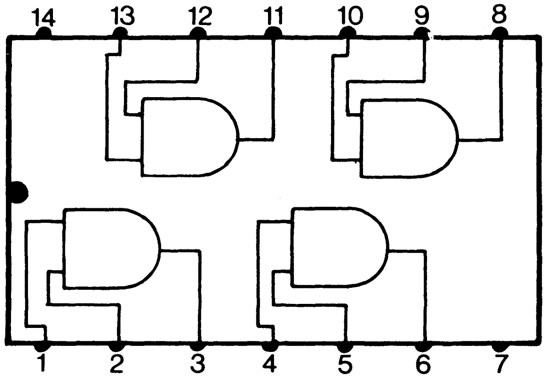


Fig. G7 A quad in line package.

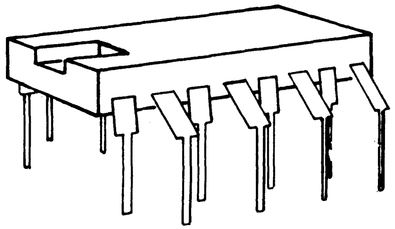
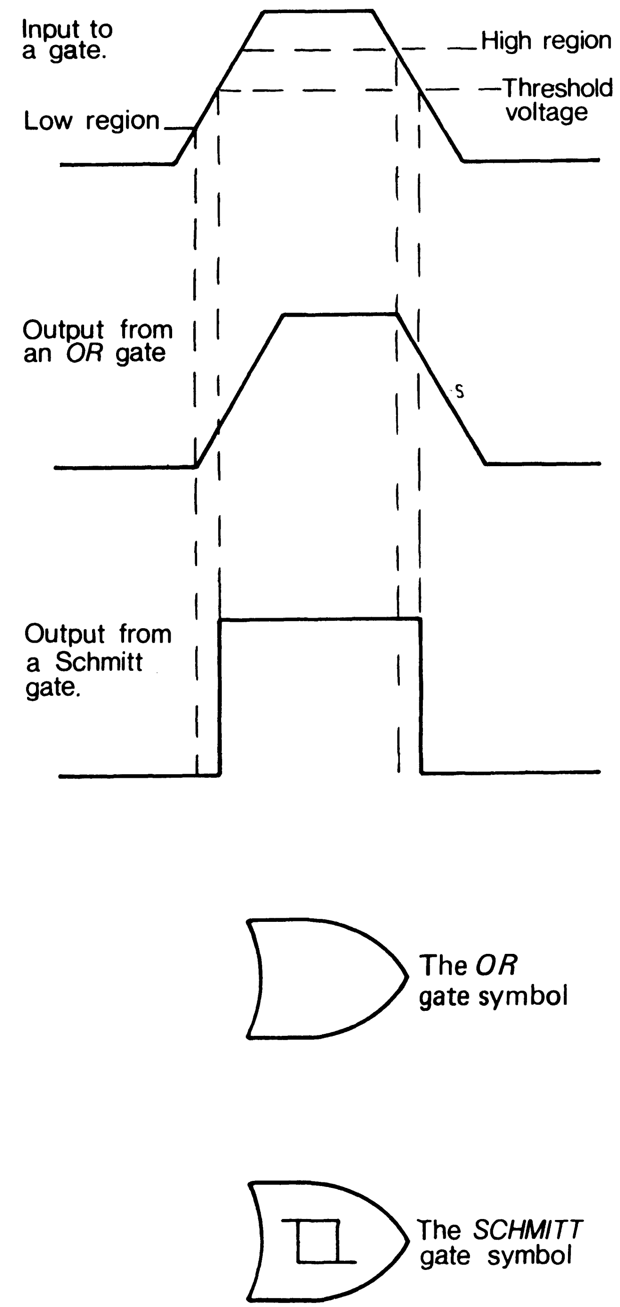


Fig. G8 Schmitt gates.



SHIFT.

The process of moving data from one place to another. Generally many bits of data are moved at once. Shifting is done synchronously and by command of the clock.

SHIFT REGISTER.

An arrangement of circuits, usually *FLIP-FLOPS,* which are used to shift serially or in parallel.

SINKING (FIGURE G9).

When the output from a gate is 'low' the output pin voltage is being pulled down to chassis by an internal output transistor. The output pin 'sinks' down to chassis.

SOURCING (FIGURE G10).

When the output from a gate is 'high', the output pin voltage is connected to HT plus by an internal output transistor. The output is being 'sourced' to the HT supply.

SSI.

SSI (Small Scale Integration), is a complete system contained in a single package. SSI could be a digital or a linear (non digital) circuit. It is a circuit that contains up to 12 gates, or circuitry of a similar complexity.

STATE.

This refers to the condition of an input or an output of a circuit. If the output pin is in the 'high' state it has a 'high' voltage on it.

SYNCHRONOUS.

Operation of a system by a clock pulse generator. All circuits in the system operate simultaneously and all actions take place in step with the clock.

SYNCHRONOUS INPUTS.

Those inputs on a *FLIP-FLOP* through which data can be entered but only upon command of the clock. These inputs do not have direct control of the output as in a logic gate. An example of this is the J-K inputs. They are sometimes called ac Set and Reset inputs.

TRIGGER.

This is a pulse used to initiate an action in a circuit.

TOTEM-POLE (FIGURE G11).

This refers to the output transistor configuration used in the TTL logic family. It gives a low impedence drive in both 'high' and 'low' output states. This is due to the fact that in the 'high' output state VT1 will be hard on, connecting the output pin to HT positive. In the 'low' output state, VT2 will be hard on, connecting the output pin down to chassis.

TTL.

The TTL (Transistor Transistor Logic) family of gates is constructed using bipolar transistors. Supply voltage 5v. Power consumption: Medium to low, depending on the series.

Two series in common use start with 74 and 54, i.e. 7400, 7401, 7402 etc. They are often called the 74 or 54 series.

Fig. G9 Sinking.

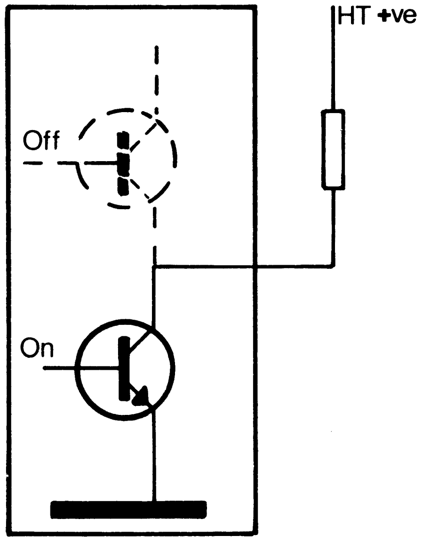


Fig. G10 Sourcing.

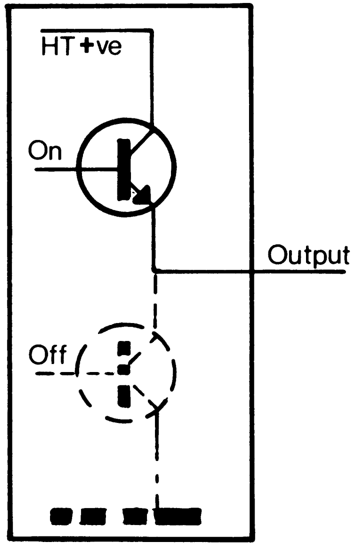
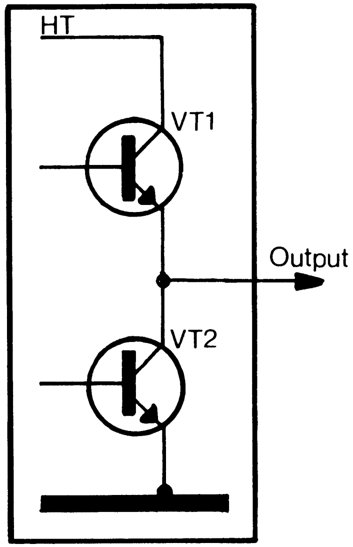


Fig. G11 Totem-pole.



WIRED LOGIC WIRED OR.

This is where the outputs of gates are connected directly together (Figure G12). The outputs can only be connected directly together when 'open collector output gates' are used.

The output is pulled down low, if gate X output or gate Y output, or both outputs, go 'low'.

The 'wired OR' situation is often drawn as shown in figure G13.

Fig. G12

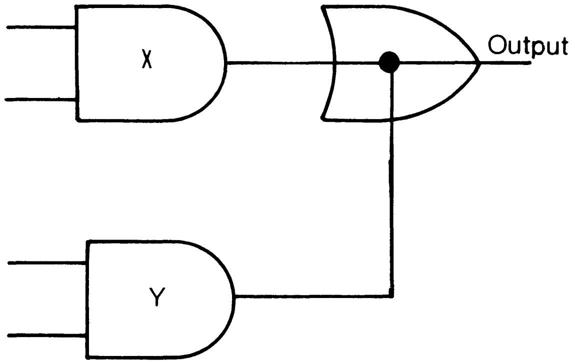
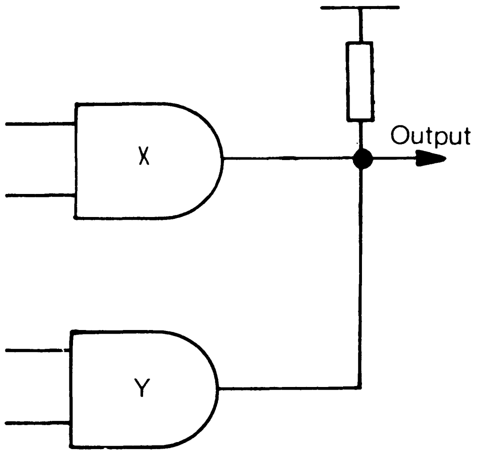


Fig. G13 The 'wired OR'.



Solutions to exercises

Page 29. Fig. 1.72

Input		Outputs	
A	B	C	X
L	L	H	L
L	H	H	L
H	L	H	L
H	H	L	H

AND gate

Page 29. Fig. 1.74

Input		Outputs			
A	B	C	D	E	X
L	L	H	H	H	L
L	H	H	H	L	H
H	L	H	L	H	H
H	H	L	H	H	L

Exclusive OR gate

Page 28. Fig. 1.71

Input				Outputs
A	B	C	D	X
L	L	L	H	H
L	H	H	H	L
H	L	H	H	L
H	H	H	L	H

Exclusive NOR

Page 29. Fig. 1.73

Inputs		Output			
A	B	C	D	E	X
L	L	H	H	L	H
L	H	H	L	H	L
H	L	L	H	H	L
H	H	L	L	H	L

NOR gate

Page 30. Fig. 1.75

Inputs		Outputs		
A	B	C	D	X
L	L	H	H	L
L	H	H	L	H
H	L	L	H	H
H	H	L	L	H

OR gate

Page 30. Fig. 1.76

Inputs		Ouputs		
A	B	C	D	X
L	L	H	H	L
L	H	H	L	L
H	L	L	H	L
H	H	L	L	H

AND gate

Page 31. Fig. 1.77

Inputs		Outputs			
A	B	C	D	E	X
L	L	H	H	L	H
L	H	H	L	L	H
H	L	L	H	L	H
H	H	L	L	H	L

NAND gate

Page 31. Fig. 1.78

Inputs		Outputs				
A	B	C	D	E	F	X
L	L	H	L	L	H	L
L	H	L	H	L	L	H
H	L	L	L	H	L	H
H	H	L	L	L	H	L

Exclusive OR gate

Index

AND function 8-9
  AND gate in toggle flip-flop 50, 79
  equivalent circuit 8
  operated by low-level input pulses 14
  practical use of AND gate 9
  voltage level truth tables 9, 15
Asynchronous counters 72-6
  block schematic representation 75
  counter as divider 76
  counting down 74
  resetting 75

Binary numbers 67-70
  voltages and binary numbers 69-70
Bistable 34
  sound mute 37
Buffer amplifier function 20
  practical example 20

Clocked 'D' type flip-flop 46
  applications 46
  circuit operation 46
  circuit symbol 46
  truth table 46
Clocked R-S flip-flop 45
  circuit operation 45
  circuit symbol 45
  truth table 45

Decade counters 86-7
  timing diagram 86
  use of 87

Edge triggered flip-flops 47-8
  D type 48, 50 (circuit operation 48; circuit symbol 47, 52)
Edge triggering 47, 48, 52

Electronic gates 3
  making NOT gate from other gates 22-3
  operated by low level input pulses 14-19
  voltage levels 3
EXCLUSIVE OR function 12-13
  control of oscillator 13
  effects of phase differences 13
  practical use for EXCLUSIVE OR gate 13
  voltage level truth table 12

Flip-flops 34
  called 'bistables' in colour TV 34
  revised for use as counters 71, 72, 78, 85
  use in representing binary numbers 69-70
  use in shift registers 88

Gate symbols 24-6, 32
  AND gate 3, 24, 26, 32 (circuit symbol 15)
  BUFFER 32
  EXCLUSIVE OR gate 25, 32 (circuit symbol 12)
  logic symbols 26
  NAND gate 10, 24, 32 (circuit symbol 10, 17)
  NOR gate 6, 25, 32 (circuit symbol 6, 19)
  NOT gate 32
  OR gate 3, 25, 26, 32 (circuit symbol 4, 18)
  state indicator circle 26

Inverting amplifier 21
  practical example 21

J-K Master Slave flip-flop 54-9
  circuit operation 55
  circuit symbol 54
  toggling 55
  truth table 54
  with set-reset facility 57 (circuit operation 57; circuit symbol 57; inputs 57; truth table 57)

Negative edge triggering 52-3
  circuit operation 53
  clock timing diagram 53
  D type flip-flop circuit symbol 52
  timing diagram 52
NOT AND function 10-11
  equivalent transistor circuit 10
  operated by low-level input pulses 16
  practical use of NAND gate 11
  use of NAND gate in decade counter 86
  used as NOT gate 23
  voltage level truth tables 11, 17
NOT OR function 6-7, 30
  operated with low-level inputs 19
  practical use for NOR gate 7
  simple equivalent circuit 6
  used as NOT gate 22
  voltage level truth tables 7, 19

OR function 4-5
  equivalent circuit 4
  operated with low-level inputs 18
  practical use of OR gate 5
  voltage level truth tables 5, 18

Prestel 92

R-S flip-flops from logic gates 38-44
  high input activated 40-1, 44, 61 (activating 40; circuit symbol 38, 40, 41, 44; dynamic operation 41; internal configuration 44; practical application 44; timing diagram 41)
  low input activated 38, 39 (activating 39; circuit symbol 38, 39, 41, 42; timing diagram 41)
  NAND gate 42 (application 43)

R-S flip-flops from logic gates (*cont.*)
  NOR gate 61, 62 (application 63; set and reset conditions 62; set and reset inputs both taken 'high' 63; set command fed to previously set flip-flop 63)

Shift registers 88-94
  PISO mode 92-3
  SIPO mode 91
  SISO mode 88, 90
  SISO/PISO mode 94

Synchronous counters 77-85
  count-down counter 81-2
  count-up counter 79-80
  count-up/down counters 83
  parallel loading 84
  TV use 84-5

Teletext 91, 94
THORN EMI remote control systems
  U703 83
  U707 and U708 83, 84
  U714 83, 84, 94
Toggle flip-flop 49-51
  applications 51
  circuit operation 50
  circuit symbol 49, 78
  truth table 49
  use in synchronous counters 78, 79
Transistor R-S flip-flop 35-7
  circuit 35